Doctoral Thesis

A Mathematical Measurement For Korean Text Mining and Its Application

Kyunghoon Kim

Department of Mathematical Sciences

Graduate School of UNIST

2018

A Mathematical Measurement For Korean Text Mining and Its Application

Kyunghoon Kim

Department of Mathematical Sciences

Graduate School of UNIST

A Mathematical Measurement For Korean Text Mining and Its Application

A dissertation
submitted to the Graduate School of UNIST
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Kyunghoon Kim

12.12.2017
Approved by

Advisor
Bong Soo Jang

A Mathematical Measurement For Korean Text Mining and Its Application

Kyunghoon Kim

This certifies that the dissertation of Kyunghoon Kim is approved.

12.12.2017

Advisor: Bongsoo Jang

Pilwon Kim : Thesis Committee Member #1

Chang Hyeong Lee : Thesis Committee Member #2

Chang-Yeol Jung : Thesis Committee Member #3

Sunghwan Moon : Thesis Committee Member #4

I dedicate this dissertation to my parents

Abstract

In modern society we are buried beneath an overwhelming amount of text data on the internet. We are less inclined to just surf the web and pass the time. To solve this problem, especially to grasp part and parcel of the text data we are presented, there have been numerous studies on the relationship between text data and the ease of the perception of the text's meaning. However, most of the studies focused on English text data. Since most research did not take into account the linguistic characters, these same methods are not suitable for Korean text. Some special method is required to analyze Korean text data utilizing the characteristics of Korean. Thus we are proposing a new framework for Korean text mining in various texts via proper mathematical measurements.

The framework is constructed with three parts:

1) text summarization
2) text clustering
3) relational text learning.

Text summarization is the method of extracting the essential sentences from the text. As a measure of importance, we propose specific formulas which focus on the characteristics of Korean. These formulas will provide the input features for the fuzzy summarization system.

However, this method has a significant defect for large data set. The number of the summarized sentences increases with the word count of a particular text. To solve this, we propose using text clustering. This field has been studied for a long time. It has a tradeoff of accuracy for speed. Considering the syllable features of Asian linguistics, we have designed 'Syllable Vector' as a new measurement. It has shown remarkable performance as implemented with text clustering, especially for high accuracy and speed through effectively reducing dimensions.

Thirdly, we considered the relational feature of text data. The above concepts deal with the document itself. That is, text information has an independent relationship between documents. To handle these relations, we designed a new architecture for text learning using neural networks (NN). Recently, the most remarkable work in natural language processing (NLP) is 'word2vec', which is built with artificial neural networks. Our proposed model has a learning structure of bipartite layers using meta

information between text data, with a focus on citation relationships. This structure reflects the latent topic of the text using the quoted information. It can solve the shortcomings of the conventional system based on the term-document matrix.

Contents

List of Figures

List of Tables

1

Introduction

A quantity of information in our world is growing exponentially. This is accelerated by a birth of WWW(World Wide Web) [1]. A feature of exponential growth is that the last term is bigger than a sum of before all of the terms. If the quantity of information of next year is bigger than information until this year, we need to handle it necessarily.

Now we are on the process toward the rapid growth, that is, the singularity is near. And the method for a lot of information will be more important. However many works only focused on English text information. It causes the leak of the method for Korean information analytics. In this work, we consider the method how to define, measure the Korean text information(including character information as a basic step) and how to solve the real-world problem with the proposed method.

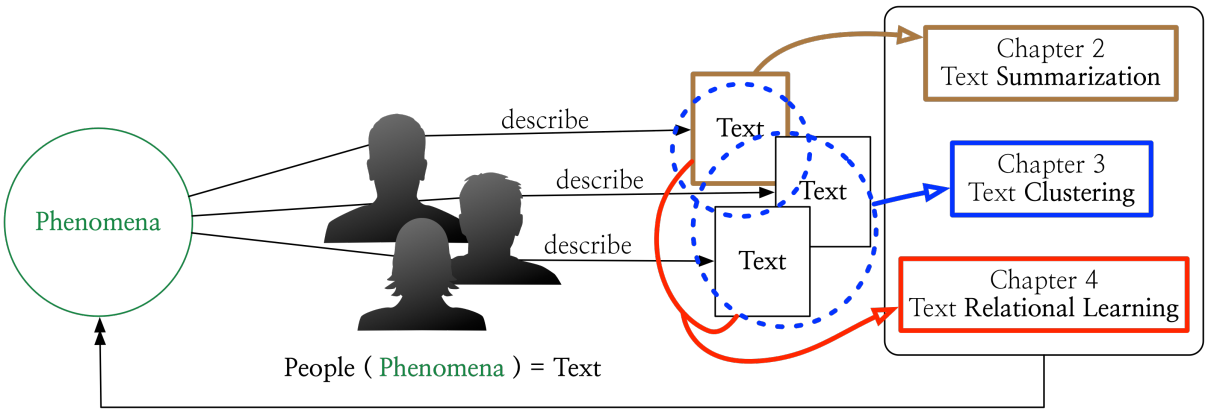


Figure 1-1: Structure of the thesis

Fundamentally, it is human to generate information. People look at a phenomenon and describe it as words in each of their own brain, with different verbal promises. Therefore we can consider that a phenomenon is an input, a brain of people is an unknown function for the processing of input information and a text is an output which is generated by a personalized function. In some ways, this can be seen as one of the inverse problems. In other words, we want to understand the essence of the phenomenon from the given text set.

The thesis consists of three central concepts. First is **text summarization**. Text summarization is to extract important sentences for one article. However, it has a clear shortage. For many of articles, extracted sentences have many volumes. Thus the next chapter is about **text clustering**. Text Clustering is to collect similar documents. Especially, we suggest the syllable vector for Korean text mining. And we propose the **text relational learning** method for meta-data of text as a final chapter.

1.1 Retrieval models for text information

Our final goal is to apply the proper mathematical modeling to an industrial problem with well-designed measurements. To solve the problem well, we need to define the design well by given data. First of all, we analyze the data, that is, we extract the data structure of given data. By this structure, we design the mathematical measurements.

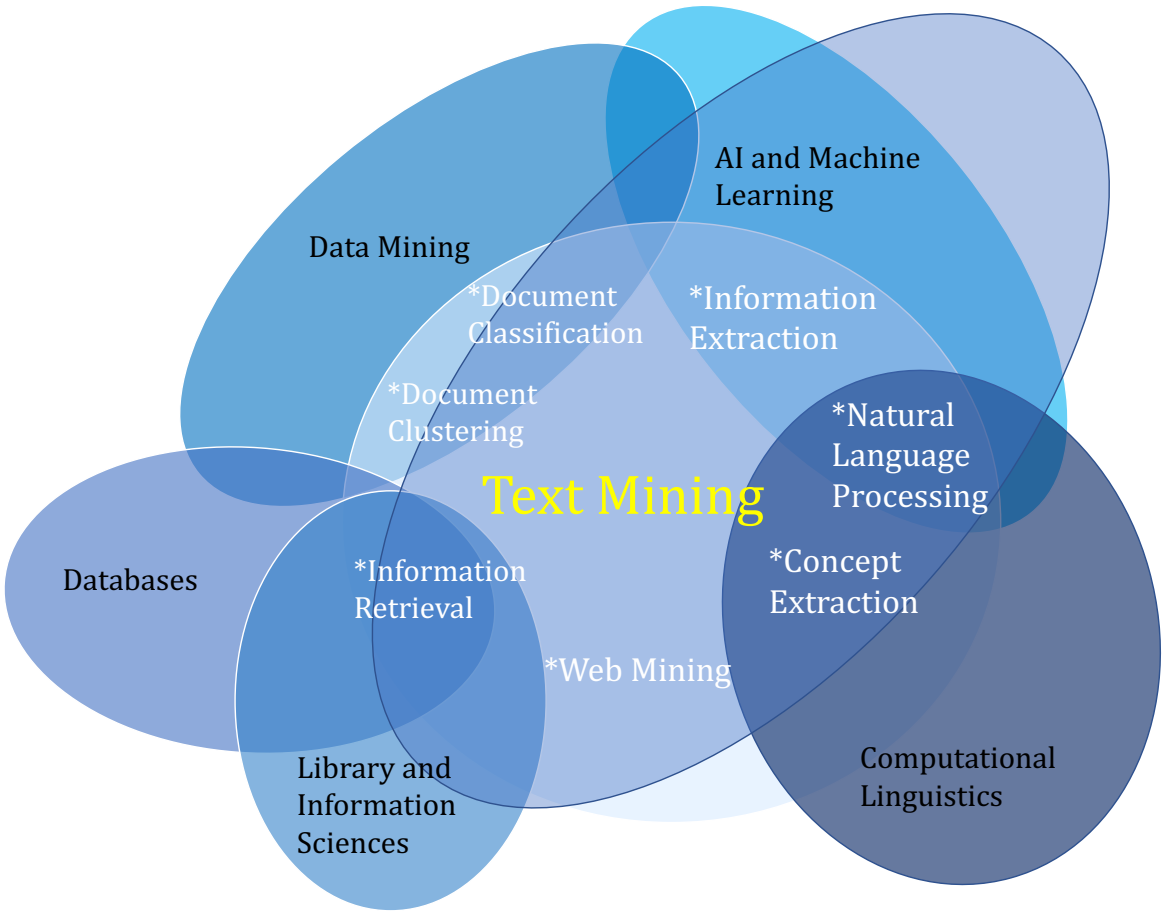


Figure 1-2: Structure of Text Mining

Text mining is a big area (Fig 1-2, [2]) and also there are a lot of the appropriate measurements to achieve their goals. In this complex area, we focus on three parts that single document, multiple documents, and relational documents.

1.1.1 Information extracting from single document

One single document has many of sentences. If we read all of that, we spend a lot of time. Thus researchers suggest the text summarizing system. A long time ago, the famous corporation *Microsoft* proposed the function "AutoSummarize(Automatically summariza a document)" in the program "word 2007"[1]. AutoSummarize identifies the key points in a document. AutoSummarize works best on well-structured documents, such as reports, articles, and scientific papers. However the accuracy of summarized data is not good and customers complain about it. So they relinquish AutoSummarize. Like this, text summarization is difficult work.

Our first central material is about summarization. We suggest the Korean text summarization system with fuzzy theory (Fig 1-3).

Figure 1-3: Text Summarization System

[1] https://support.office.com/en-us/article/Automatically-summarize-a-document-B43F20AE-EC4B-41CC-B40A-753EED

1.1.2 Information extracting from multi-documents

One of the work for the information extracting from the multi-documents is to make the relationship between documents. The documents that are given without any relation are fragments of information to be investigated individually. If the documents are grouped according to the similarity of the content, we can easily obtain the information as many as the number of groups. We call to the above-mentioned data as *unstructured data*, and the latter as *structured data* (Fig 1-4).

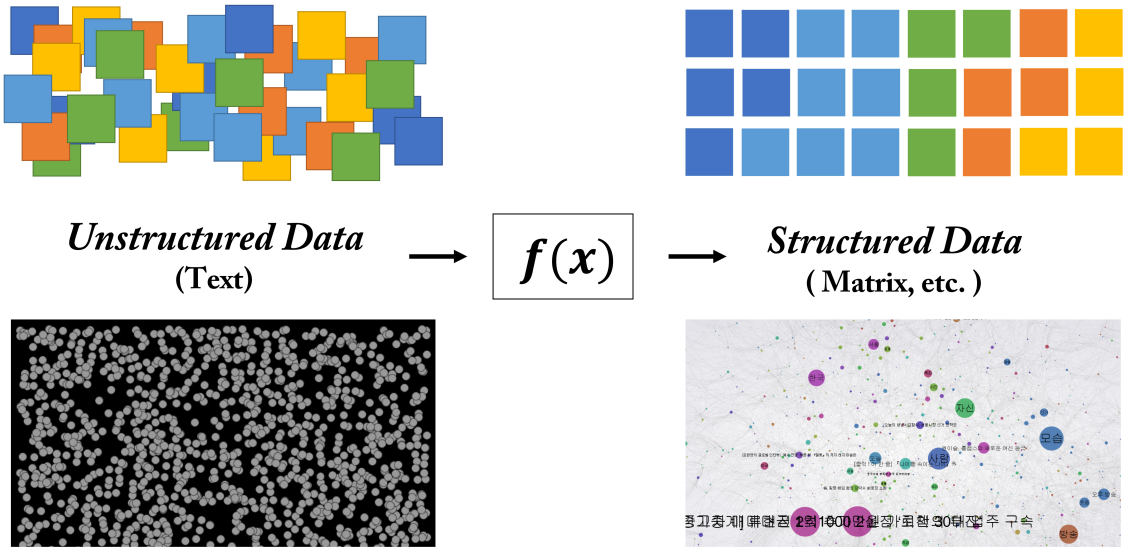


Figure 1-4: Manufacturing from unstructured data to structured data

One of the reasons we need to find out the relationship of the documents is as follows. In the news industry, there is a lot of news buzzing(abusing) as a way to attract traffic to earn the advertising money (Fig 1-5). These articles differ only slightly in sentences or titles. This causes unnecessary inundation of information. However when we apply the clustering method to it, we can neatly sort out the information. In the Figure 1-6, the points indicate one news article, and lines indicate an association. According to any arbitrary value, this is separated, and each group represents one piece of information.

	title	corp	realtime
66	MBC 연기대상 최민수 수상거부 "법과 상식 무너지고 진실과 양심 박제된..."	donga	2014-12-31 11:21:00
163	MBC 연기대상 최민수, 황금 연기상 수상거부 왜? "잘한 게 있어야 상을 받죠"	donga	2014-12-31 11:07:00
377	MBC 연기대상 최민수 수상거부 "아직도 차가운 바다 깊숙이 갇혀 있는..."	donga	2014-12-31 10:41:00
488	MBC 연기대상 최민수 수상거부 "진실과 양심이 박제된 이 시대에..."	donga	2014-12-31 10:26:00
529	MBC 연기대상 최민수 불참, 백진희가 대신 '수상거부' 소식 전해...이유는?	donga	2014-12-31 10:20:00
617	MBC 연기대상 최민수, 이유 있는 수상거부 "차가운 바다 깊숙이 갇혀있는..."	donga	2014-12-31 10:08:00
740	MBC 연기대상 최민수, 불참에 수상거부까지? "받을 게 뭐가 있겠나"	donga	2014-12-31 09:52:00
879	최민수, 'MBC 연기대상' 황금 연기상 수상거부 이유는? 소감 전문 보니...	donga	2014-12-31 09:35:00
996	'최민수 연기대상' 최민수 수상거부 "뭐 잘한 게 있어야 상을 받죠. 그죠?"	donga	2014-12-31 09:18:00

Figure 1-5: Problem of News abusing

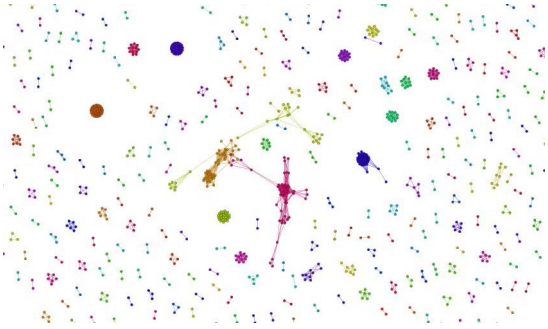


Figure 1-6: News Clustering by the content similarity

There have been various proposals for clustering methods, but there are significant limitations to such methods. Especially, word-based methods have a very large matrix size depending on the number of words. To solve this problem, we propose a new method *Syllable Vector* for the clustering of Korean text and compare the performance with the existing method to show how efficient the method is.

1.1.3 Information extracting from relational documents

Many traditional methods to understand the contents of a document are based on words. If it is the documents whose content is not analyzed on a word-by-word basis, document analysis will inevitably fail. Thus we tried a completely new approach to analyzing content based on relationships between documents.

First, we assume two kinds of data sets and show the results through a proposed method *Heterogeneous Word2vec* how the associations are well learned when the sets have relations. And we apply this method to legal data and show that the performance was excellent.

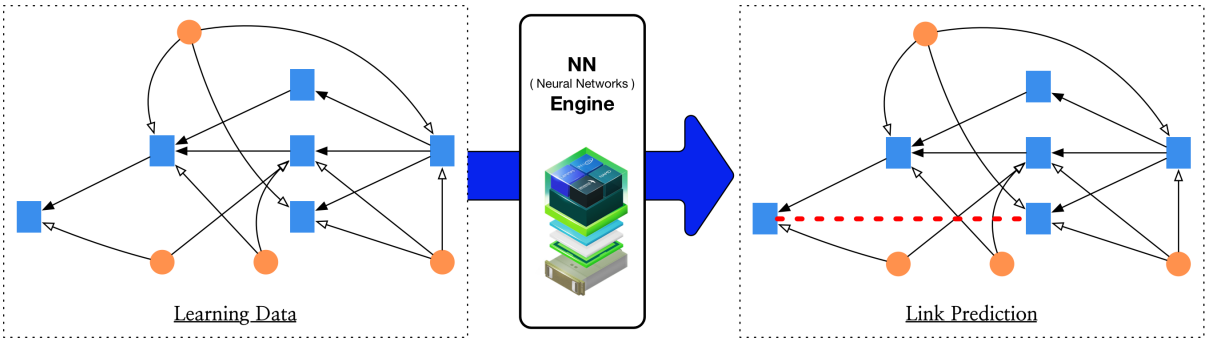


Figure 1-7: Neural Network Engine with learning data

2

Korean News Summarization using Fuzzy Logic

Summarization is classified in two ways, extraction and abstraction. In this chapter, the summarization falls under the *extraction* classification. Summarization is like human-behavior. Deciding on the importance of each sentence is the difficult from the human point of view. There are many factors in deciding whether each sentence is important or not. There is an ambiguity when making this evaluation using fuzzy logic.

2.1 Motivation

If you have some interest for IT, you have had a familiar ring about 'Summly'. Yahoo spent a reported $30 million on Summly in 2013, this is a mobile app for a teenager's 5-month-old app. We wondered about the value of this technic when we heard this news.

We were excited about it really for two primary reasons. First, summarization technology can make our life much more simple. Today's society has so much information, there is exponential growth and then from that comes complexity. The second reason is that his age is only seventeen. We can do that if he can do it.

Nowadays we have various information issues, and then we can be overcome by clever design. So we expected that this is part of the loom connecting the clever design.

2.2 Fuzzy Logic

The concept of a fuzzy set represented by its membership function which was introuced by Zadeh in 1965[3]. The main feature of fuzzy logic is that it is able to deal with imprecise linguistic information. Clearly, "the class of big number" or "the class of handsome guys" do not consist of

class in the mathematical sense. In the fuzzy sense, this ambiguity is a "class" with a continuum of grades of membership. The notion of a fuzzy set provides a convenient point of this difficulty.

Definition 2.2.1 (Fuzzy set)**.** *Let X be a space of points(objects), with a generic element of X denoted by x. A* fuzzy set *A in X is characterized by a* membership function *$f_A(x)$ which associates with each point in X a real number in the interval $[0, 1]$, with the value of $f_A(x)$ at x representing the "grade of membership" of x in A.*

Definition 2.2.2. *A is* empty set *if and only if $f_A(x) = 0$ for all $x \in X$. A and B are* equal *if and only if $f_A(x) = f_B(x)$ for all $x \in X$.* emplement *of A is a fuzzy set A' such that $f_{A'}(x) = 1 - f_A(x)$. A is* contained *in B(A is a subset of B) if and only if $f_A(x) \leq f_B(x)$.* Union *of A and B is the smallest fuzzy set containing both A and B such that $f_{A\cup B}(x) = max(f_A(x), f_B(x))$.* Intersection *of A and B is the largest fuzzy set containing both A and B such that $f_{A\cap B}(x) = min(f_A(x), f_B(x))$.*

Definition 2.2.3 (three-valued of Kleene)**.** *The notion of 'belonging', which plays a fundamental role in the case of ordinary sets, does no have the same role in the case of fuzzy sets.*

1. *x belong to A if $f_A(x) \geq \alpha$*
2. *x does not belong to A if $f_A(x) \leq \beta$*
3. *x has an indeterminate status relative to A if $\beta < f_A(x) < \alpha$*

Definition 2.2.4 (Algebraic operations on fuzzy sets)**.** *The* algebraic product *of A and B is denoted by AB and is defined in terms of the membership functions of A and B by the relation $f_{AB} = f_A f_B; AB \in A \cap B$. The* algebraic sum *of A and B is denoted by $A + B$ and is defined by $f_{A+B} = f_A + f_B$. Unlike the algebraic product, the algebraic sum is meaningful only when the condition $f_A(x) + f_B(x) \leq 1$ is satisfied for all x. The* absolute difference *of A and B is denoted by $|A - B|$ and is defined by $f_{A-B} = |f_A - f_B|$.*

Definition 2.2.5 (Convex combination)**.** *Let A, B, and Λ be arbitrary fuzzy sets. The convex combination of A, B, and Λ is denoted by $(A, B; \Lambda)$ and is defined by the relation $(A, B; \Lambda) = \Lambda A + \Lambda' B$ where Λ' is the complement of Λ.*

This can be as membership functions, $f_{(A,B;\Lambda)}(x) = f_\Lambda(x) f_A(x) + (1 - f_\Lambda(x)) f_B(x), x \in X$. Especially, $A \cap B \subset (A, B; \Lambda) \subset A \cup B$ for all Λ.

And there are the following concepts : *fuzzy relation*(a natural extension of fuzzy sets), *convexity*($\Gamma_\alpha = \{x | f_A(x) \geq \alpha\}$), *boundedness*(all α-cuts are bounded), and so on.

2.3 Summarization

The document of ISO 215 standards in 1986 formally defines a summary as a "brief restatement within the document (usually at the end) of its salient findings and conclusions" that "is intended to complete the orientation of a reader who has studied the preceding text." [6] Our news summarization is a automatic text summarization using the technic in which a computer automatically creates such a summary.

2.4 Survey of Summarization

There are many research in the summarization area. Especially, in the fuzzy area, in the big point of view, there are two kinds of approach. The first is to make new sentence. This is a ***abstraction***. Second, to find the most significant sentences, it is a ***extraction***. When I have a presentation in the class, that is a type of abstraction in sense of Yager[4]. In this paper the type of summarization is a extraction. The technic of Summly is more of extraction than abstraction.

The next generation is, in my opinion, a kind of hybrid, i.e., the compound of abstraction and extraction. This work will need the new concept like the formation of Yager, Kacprzyk[5].

2.5 System Framework

In this section, I propose the framework for the summarization. This framework based on morpheme using corpus and fuzzy logic is the main part of this work.

Fuzzy inference methods are classified in direct methods and indirect methods. The type like Mamdani and Sugeno is the kind of direct method. Mamdani fuzzy inference system(FIS) is the most commonly used in applications, due to its simple structure of min-max operations.[10] We use the type of mamdani.

2.5.1 Source Document

There are many RSS(Really Simple Syndication) and website of newspaper. We can save this contents to database, i.e., crawl(a algorithm to automatically download web pages).

2.5.2 Preprocessing

The parser performed the work which is the removing tag, advertising sentences and images. We can do this using the build-in function of Programming Language(e.g., replaced, split).

2.5.3 Extraction of Features

In the summarization technic, the core part is the feature measurement. The result is different depending on what we select. Then from what I search, there are almost fourteen features.[8]

(a) Content word(Keyword) feature
Content words are usually nouns and determined using tf-idf measure. **tf-idf**, term frequency-inverse document frequency, is a numerical statistic which reflects how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining.[9]

(b) Title word feature
Title have some important keyword. Then we calculate the importance of a sentence.

(c) Sentence location feature
For instance, the location is forepart in document. First and last sentence is usually important.

(d) Sentence Length feature
In ordinary case, very large and very short sentences are not included in summary.

(e) Proper Noun feature
In this feature, the proper noun is name of something.

(f) Upper-case word feature
acronym or proper noun.

(g) Cue-Phrase feature
E.g., 'in conclusion', 'finally', ...

(h) Biased Word feature
In the some case, the sentence including biased word is important.

(i) Font based feature
The font of bold or italics is special thing.

(j) Pronouns
E.g., she, they, it. This is not important as summary.

(k) Sentence-to-Sentence Cohesion
This is part of calculating the similarity.

(l) Sentence-to-Centroid Cohesion
Before calculate of sentence-to-sentence cohesion, as a arithmetic average we use this.

(m) Occurrence of non-essential information
E.g., 'because', 'furthermore', 'additionally'.

(n) Discourse analysis
This have something to do with discourse level information. It is necessary to determine the overall discourse structure of the text.

However the existing feature measurement is suitable for english sentences. Thus we suggest the proper features for Korean. We select the features that 'Ratio between title and content' and 'Ratio of part of speech'. Each values are as input values in Fuzzy Inference System(FIS).

2.5.3.1 Ratio between title and content (V1)

We find the nouns in the title and the content using a morpheme analyser, respectively. The ratio is that

$$V1 = w \times \frac{A}{B},$$

where w is a weight of the measure, A is a number of equal nouns and B is a number of total nouns in a content. We use $w = 1.5$.

2.5.3.2 Ratio of Part of Speech (V2)

A number of Part Of Speech, especially auxiliary marker(JX) and Adverbial case markers(JKB), is measured.

$$V2 = \frac{C + D}{E},$$

where C is a number of JX, D is a number of JKB and E is a total nouns in the content.

2.5.4 Text Summarization based on Fuzzy Logic

The fuzzy logic system consists of four components: fuzzifier, inference engine, defuzzifier, and the fuzzy knowledge base. In the fuzzifier, crisp inputs are translated into linguistic values using a membership function to be used to the input linguistic variables. After fuzzification, the inference engine refers to the rule base containing fuzzy IF-THEN rules to derive the linguistic values. In the last step, the output linguistic variables from the inference are converted to the final crisp values by the defuzzifier using membership function for representing the final sentence score.[7] Then we get finished with this process2.5.3, i.e, calculation of sentence score.

2.5.5 Refinement

The sentences are sorted by weights. The high score is a most important sentence. Usually, we are aided by ordering of database system.

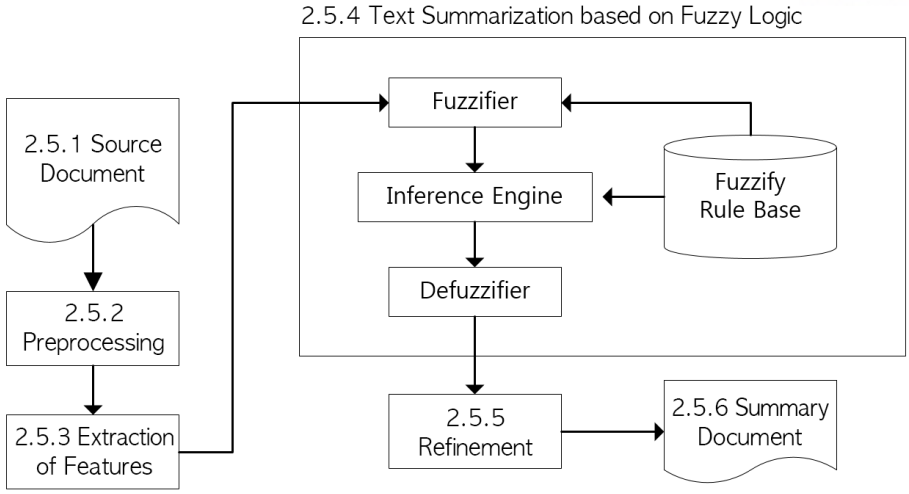


Figure 2-1: Flowchart of System Framework

2.5.6 Summary Document

We display the 'title', 'url', 'date' and align the three sentences by rank. We called it the summary of document.

2.6 Impremention

There exists a few fuzzy libraries on python: Pyfuzzy, Fuzzypy, Peach Fuzzy, Gfuzzy[11]. In order to implement we use a library "Peach.v0.3.1". In this section, let's take a look at this functions briefly. [12]

2.6.1 Fuzzy Sets and Fuzzy Operators

```
>>> a=peach.FuzzySet([0,0.25,0.5,0.75,1])
>>> b=peach.FuzzySet([0,1,0,1,0])
>>> a&b % and operator
FuzzySet([ 0. , 0.25, 0.  , 0.75, 0. ])
>>> a|b % or operator
FuzzySet([ 0. , 1. , 0.5, 1. , 1. ])
>>> -a % not operator
FuzzySet([-0. , -0.25, -0.5 , -0.75, -1. ])
>>> -b
FuzzySet([-0., -1., -0., -1., -0.])
```

Listing 2.1: PYTHON implementation of Fuzzy Sets

2.6.2 Membership Functions

- peach.**Triangle(x0, x1, x2)**
  - A triangle function, returning 0 if x is less than x0 of greater than x2, a maximum value of 1 if x is equal to x1 and straight lines connecting these points. Notice that x0 must be lower than x1, and that both must be lower than x2.
- peach.**Trapezoid(x0, x1, x2, x3)**
  - A trapezoid function, returning 0 if x is less than x0 of greater than x3, a value of 1 if x is between x1 and x2 and straight lines connecting these points. Notice that we must assure that x0 $<$ x1 $<$ x2 $<$ x3.

For example, to create a triangle function starting at 0, with peak in 1, and ending in 2, then this is it:

```
>>> mb = peach.Triangle(0,1,2)
>>> mb(1)
FuzzySet(1.0)
>>> mb(1.5) % fuzzification
FuzzySet(0.5)
```

Listing 2.2: PYTHON implementation of a Triangle function and Fuzzification

2.6.3 Defuzzification

In the Mamdani-type fuzzy inference system, we use the centre of area(Centroid) for the defuzzification process.

```
>>> y = numpy.linspace(0,5,100) % assign the domain values
>>> m_y = peach.Triangle(1,2,3) % triangle membership function
>>> print Centroid(m_y(y),y)
2.00010307153
```

Listing 2.3: PYTHON implementation of Defuzzification

- peach.fuzzy.defuzzy.**Centroid(mf, y)**
  - Center of gravity method. The center of gravity is calculate using the standard formula found in any calculus book. The integrals are calculated using the trapezoid method.
  - **Parameters**

* **mf**: Fuzzy set containing the membership values of the elements in the vector given in sequence
* **y**: Array of domain values of the defuzzified variable.

2.6.4 Korean News Summarization

For implementation, we see the following example. The code is the Fuzzy Inference Function.

```
g=Gnuplot.Gnuplot(debug=1)
g.title('Aggregation of the implied output fuzzy sets')
g.xlabel('Value of the sentence')
g.ylabel('membership')
g('set term png')
g('set out "output.png"')

% membership functions
y=numpy.linspace(0,1,1000)
i_zero = peach.Triangle(-0.1,0,0.1)
i_low = peach.Triangle(0,0.25,0.5)
i_medium = peach.Triangle(0.2,0.5,0.8)
i_high = peach.Trapezoid(0.5,0.8,1,1.1)
o_zero,o_low,o_medium,o_high,o_veryhigh = peach.FlatSaw((0,1),5)
Points = 100
yrange = numpy.linspace(0.,1.,1000)
c = peach.Controller(yrange)

% Fuzzy Rules
c.add_rule(((i_zero,i_zero), o_zero))
c.add_rule(((i_zero,i_low), o_low))
c.add_rule(((i_zero,i_medium), o_medium))
c.add_rule(((i_zero,i_high), o_medium))
c.add_rule(((i_low,i_zero), o_low))
c.add_rule(((i_low,i_low), o_medium))
c.add_rule(((i_low,i_medium), o_medium))
c.add_rule(((i_low,i_high), o_high))
c.add_rule(((i_medium,i_zero), o_medium))
c.add_rule(((i_medium,i_low), o_medium))
c.add_rule(((i_medium,i_medium), o_medium))
c.add_rule(((i_medium,i_high), o_high))
```

```
c.add_rule(((i_high,i_zero), o_medium))
c.add_rule(((i_high,i_low), o_high))
c.add_rule(((i_high,i_medium), o_high))
c.add_rule(((i_high,i_high), o_veryhigh))
c.defuzzy = peach.Centroid

x = numpy.linspace(0,1,Points)
y = []
for x0 in x:
y.append(c(x0)) % defuzzification value
y=numpy.array(y)

d1 = Gnuplot.Data(x,y,with_="line")
g('set yrange [0:1]')
g.plot(d1)
```

Listing 2.4: PYTHON implementation of Fuzzy inference function

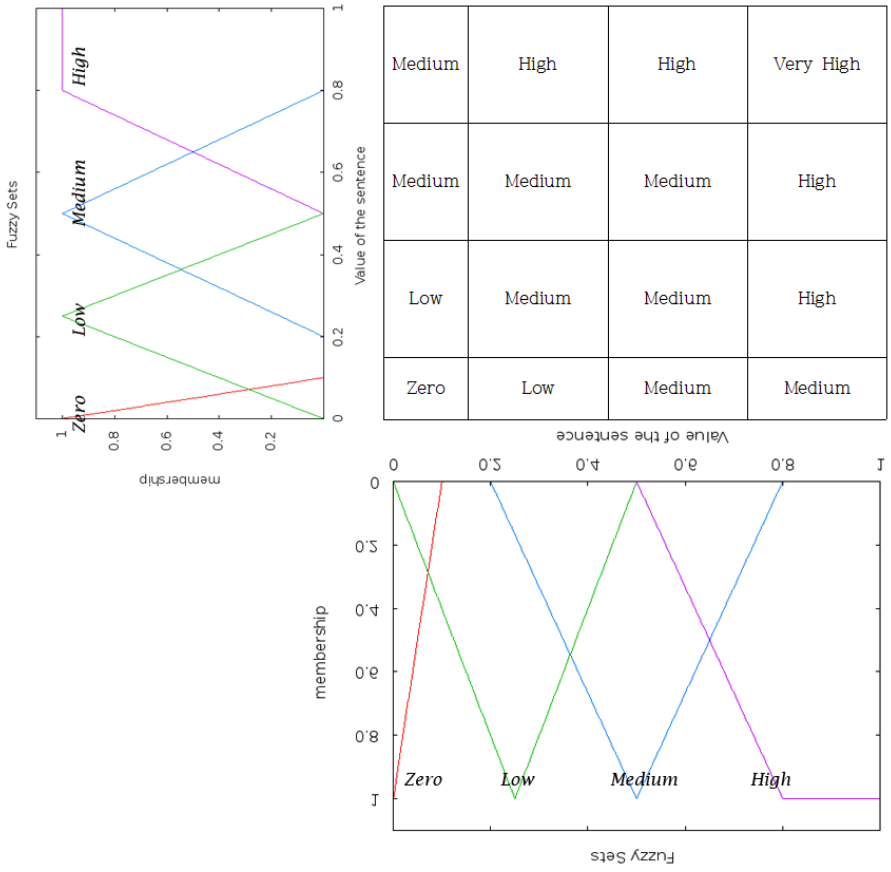


Figure 2-2: Rule matrix and Fuzzy sets of V1, V2

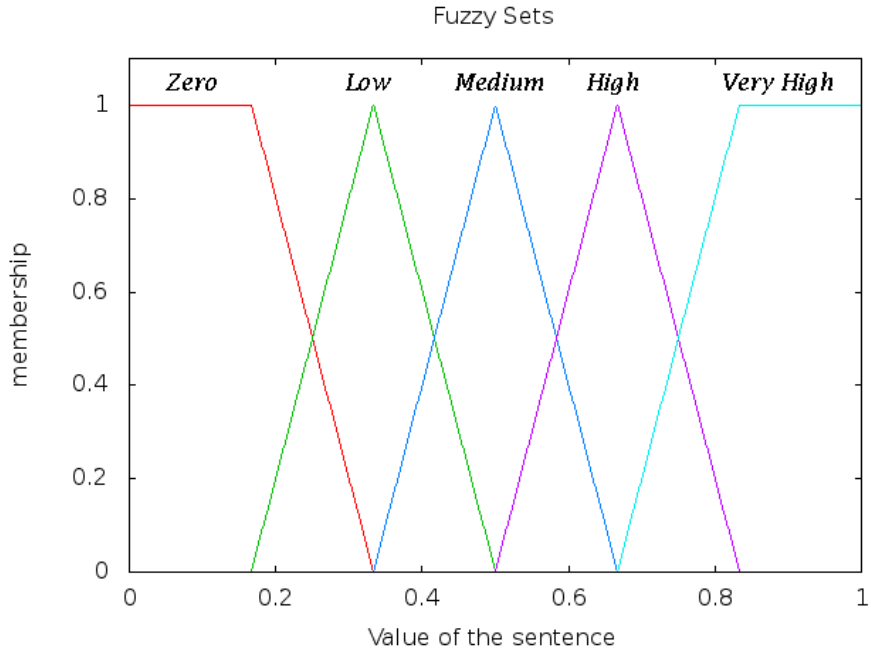


Figure 2-3: Fuzzy sets of output

Then we assume that following article. This article is refined like tags by preprosessing.

subject : 아마존, 환각물질에 흡입도구까지 버젓이
content : (지디넷코리아)아마존에서 환각물질과 마약성 물질 흡입용 파이프와 물담뱃대가 버젓이 팔리고 있다.영국 데일리메일은 12일 가스로 알려진 아질산 등을 버젓이 팔고 있다고 보도했다. 환각약품 흡입용인 파이프와 물담뱃대같은 도구도 함께 팔리고 있었다. 보도는 이베이같은 쇼핑사이트에서는 이런 (합법적인)환각제나 환각성 약품 및 기구를 팔지 않지만 아마존에서는 팔리고 있다고 전했다. ▲ 아마존이 강력한 환각성분을 가진 GBL이나 환각제 흡입도구인 물담뱃대,파이프 등을 팔고 있어 비난을 사고 있다. 이베이는 이를 팔고 있지 않다.<사진=아마존>▲ 영국의 언론들은 아마존이 아질산이나 샐비아(사진)같은 환각성 제품을 버젓이 팔고 있다며 이는 이베이같은 곳에서는 이뤄지지 않는 일이라고 전했다. <사진=데일리메일>데일리메일은 특히 딸이 아마존에서 구입한 솔벤트용제 GBL(gamma-Butyro Llactone)을 구입해 흡입하다 사망하자 아마존에 편지를 쓴 한 어머니의 사연도 소개했다. 보도는 이 어머니가 아마존에 GBL 판매중지를 요청했지만 아무런 답도 받지 못했다고 전했다. 그녀는 인터뷰에서 “아마존은 이같은 물품 판매에 대해 아주 무책임하며, 이는 환각물품 판매를 지지하는 것과 마찬가지”라고 비난했다. 이 어머니는 또 ”아마존은 믿을 수 있는 브랜드지만 뭔가를 팔려고 할 때에는 젊은이들에게 뭔가가 안전하게 믿도록 오도하려는 듯이 보인다”고 꼬집었다. 또 “우리는 이들 제품이 안전하지 않으며 사람들이 이를 취하는 것은 그들의 생명과 정신건강을 볼모로 러시안 룰렛을 하는 것과 같다는 걸 안다 ”고 말했다. 이밖에도 아마존은 존 스트롱이라는 회사가 공급하는 강력한 환각 성분이 들어있는 것으로 알려진 샐비아도 판매하고 있었다. 이 회사는 샐비아가 LSD보다 더 강력하고 합법적이라고 광고하고 있다. 일부 고객들은 이것이 마약으로

사용될 수 있다고 말한다. 아마존 대변인은 “모든 아마존 판매자들은 우리웹사이트에서 판매되는 제품에 대해 우리의 가이드라인을 준수해야 한다. 우리는 이를 준수하도록 다양한 방식을 사용하고 있다”고 말했다. 그는 “우리는 가이드라인에 저촉되는 모든 물품을 즉각 제거하고 있으며 문제가 되는 판매자들에 대해 적절한 조치를 취하고 있다”고 주장했다. ▶ IT 세상을 바꾸는 힘 지디넷코리아,▶ IT뉴스는 지디넷코리아, 게임트렌드는 게임스팟코리아▶ 스마트폰으로 읽는 실시간 IT뉴스 모바일지디넷 저작권자 메가뉴스 & ZDNet & CNET. 무단전재 및 재배포 금지

The first measure is ‘ratio between title and content’(V1). And second measure is ‘ratio of part of speech’(V2). Then each sentence have a set of values. $[V1, V2/defuzzification]$. (The weight of V1 is 1.5)

‘아마존에서 환각물질과 마약성 물질 흡입용 파이프와 물담뱃대가 버젓이 팔리고 있다 $[0.666666666667, 0.111111111111/\mathbf{0.583333627399}]$

‘영국 데일리메일은 12일(현지시간) 아마존이 LSD보다 더 강력한 환각작용을 가진 솔벤트성분 화학용제, 웃음가스로 알려진 아질산 등을 버젓이 팔고 있다고 보도했다 $[0.1875, 0.1875/\mathbf{0.5}]$

환각약품 흡입용인 파이프와 물담뱃대같은 도구도 함께 팔리고 있었다 $[0.642857142857, 0.142857142857/\mathbf{0.580158190811}]$

보도는 이베이같은 쇼핑사이트에서는 이런 (합법적인)환각제나 환각성 약품 및 기구를 팔지 않지만 아마존에서는 팔리고 있다고 전했다 $[0.3, 0.5/\mathbf{0.5}]$

▲ 아마존이 강력한 환각성분을 가진 GBL이나 환각제 흡입도구인 물담뱃대,파이프 등을 팔고 있어 비난을 사고 있다 $[0.6, 0.0/\mathbf{0.5}]$

<사진=아마존>▲ 영국의 언론들은 아마존이 아질산이나 샐비아(사진)같은 환각성 제품을 버젓이 팔고 있다며 이는 이베이같은 곳에서는 이뤄지지 않는 일이라고 전했다 $[0.230769230769, 0.307692307692/\mathbf{0.5}]$

<사진=데일리메일>데일리메일은 특히 딸이 아마존에서 구입한 솔벤트용제 GBL(gamma-Butyro Llactone)을 구입해 흡입하다 사망하자 아마존에 편지를 쓴 한 어머니의 사연도 소개했다 $[0.1875, 0.25/\mathbf{0.5}]$

보도는 이 어머니가 아마존에 GBL 판매중지를 요청했지만 아무런 답도 받지 못했다고 전했다 $[0.214285714286, 0.428571428571/\mathbf{0.5}]$

그녀는 인터뷰에서 “아마존은 이같은 물품 판매에 대해 아주 무책임하며, 이는 환각물품 판매를 지지하는 것과 마찬가지”라고 비난했다 $[0.272727272727, 0.454545454545/\mathbf{0.5}]$

이 어머니는 또 ”아마존은 믿을 수 있는 브랜드지만 뭔가를 팔려고 할 때에는 젊은이들에게 뭔가가 안전하게 믿도록 오도하려는 듯이 보인다”고 꼬집었다 $[0.214285714286, 0.714285714286/\mathbf{0.612995568351}]$

이밖에도 아마존은 존 스트롱이라는 회사가 공급하는 강력한 환각 성분이 들어있는 것으로 알려진 샐비아도 판매하고 있었다 $[0.375, 0.625/\mathbf{0.572161105654}]$

아마존 대변인은 “모든 아마존 판매자들은 우리웹사이트에서 판매되는 제품에 대해 우리의 가이드라인을 준수해야 한다 [0.166666666667, 0.444444444444/**0.5**]

So the best sentences are following:

- 이 어머니는 또 “아마존은 믿을 수 있는 브랜드지만 뭔가를 팔려고 할 때에는 젊은이들에게 뭔가가 안전하게 믿도록 오도하려는 듯이 보인다”고 꼬집었다 [0.214285714286, 0.714285714286/**0.612995568351**]
- ‘아마존에서 환각물질과 마약성 물질 흡입용 파이프와 물담뱃대가 버젓이 팔리고 있다 [0.666666666667, 0.111111111111/**0.583333627399**]
- 환각약품 흡입용인 파이프와 물담뱃대같은 도구도 함께 팔리고 있었다 [0.642857142857, 0.142857142857/**0.580158190811**]

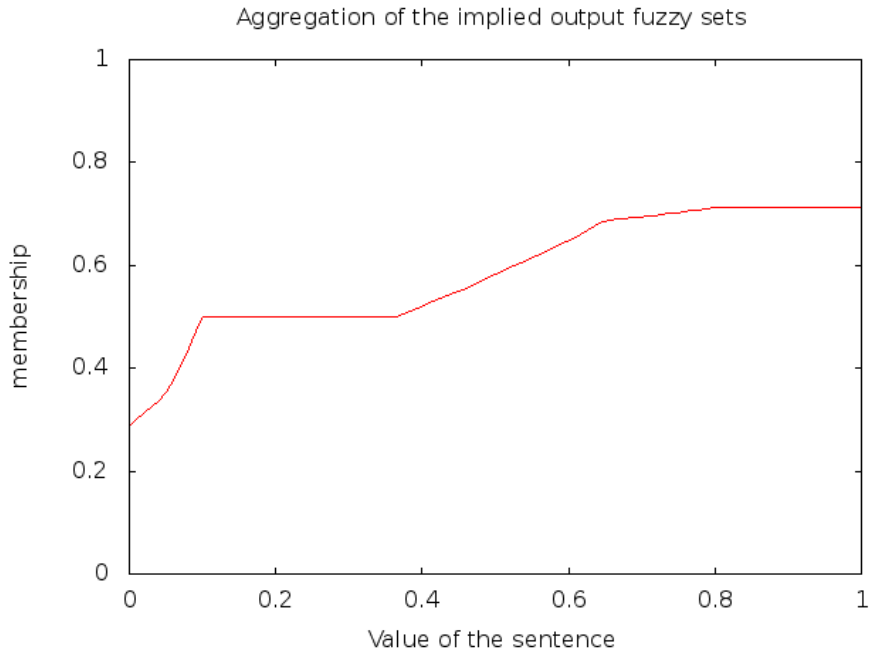
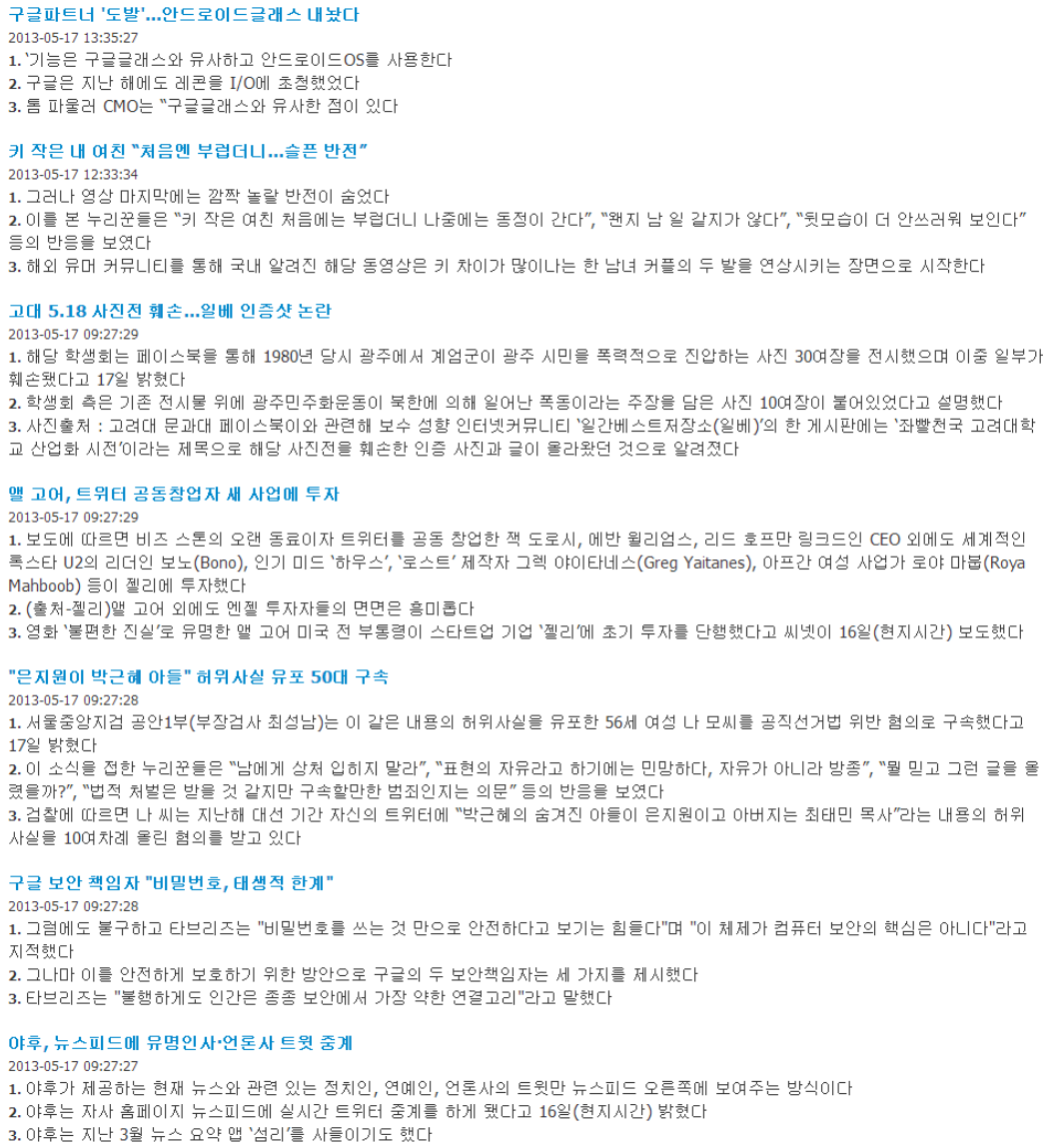


Figure 2-4: Defuzzification values with membership function

Then we can apply to latest news. Auto generating summarization news list is the following frame box. We can experience it in the url `http://mathenary.net/news`. Look through the below result list, then feel that you can understand the news content before read the whole news.

크라이텍, 넥슨 통해 워페이스 일본 공략
2013-05-17 14:06:04



구글파트너 '도발'...안드로이드글래스 내놨다
2013-05-17 13:35:27
1. '기능은 구글글래스와 유사하고 안드로이드OS를 사용한다
2. 구글은 지난 해에도 레콘을 I/O에 초청했었다
3. 톰 파울러 CMO는 "구글글래스와 유사한 점이 있다

키 작은 내 여친 "처음엔 부럽더니...슬픈 반전"
2013-05-17 12:33:34
1. 그러나 영상 마지막에는 깜짝 놀랄 반전이 숨었다
2. 이를 본 누리꾼들은 "키 작은 여친 처음에는 부럽더니 나중에는 동정이 간다", "왠지 남 일 같지가 않다", "뒷모습이 더 안쓰러워 보인다" 등의 반응을 보였다
3. 해외 유머 커뮤니티를 통해 국내 알려진 해당 동영상은 키 차이가 많이나는 한 남녀 커플의 두 발을 연상시키는 장면으로 시작한다

고대 5.18 사진전 훼손...일베 인증샷 논란
2013-05-17 09:27:29
1. 해당 학생회는 페이스북을 통해 1980년 당시 광주에서 계엄군이 광주 시민을 폭력적으로 진압하는 사진 30여장을 전시했으며 이중 일부가 훼손됐다고 17일 밝혔다
2. 학생회 측은 기존 전시물 위에 광주민주화운동이 북한에 의해 일어난 폭동이라는 주장을 담은 사진 10여장이 붙어있었다고 설명했다
3. 사진출처 : 고려대 문과대 페이스북이와 관련해 보수 성향 인터넷커뮤니티 '일간베스트저장소(일베)'의 한 게시판에는 '좌빨천국 고려대학교 산업화 시전'이라는 제목으로 해당 사진전을 훼손한 인증 사진과 글이 올라왔던 것으로 알려졌다

앨 고어, 트위터 공동창업자 새 사업에 투자
2013-05-17 09:27:29
1. 보도에 따르면 비즈 스톤의 오랜 동료이자 트위터를 공동 창업한 잭 도로시, 에반 윌리엄스, 리드 호프만 링크드인 CEO 외에도 세계적인 록스타 U2의 리더인 보노(Bono), 인기 미드 '하우스', '로스트' 제작자 그렉 야이타네스(Greg Yaitanes), 아프간 여성 사업가 로야 마붑(Roya Mahboob) 등이 젤리에 투자했다
2. (출처-젤리)앨 고어 외에도 엔젤 투자자들의 면면은 흥미롭다
3. 영화 '불편한 진실'로 유명한 앨 고어 미국 전 부통령이 스타트업 기업 '젤리'에 초기 투자를 단행했다고 씨넷이 16일(현지시간) 보도했다

"은지원이 박근혜 아들" 허위사실 유포 50대 구속
2013-05-17 09:27:28
1. 서울중앙지검 공안1부(부장검사 최성남)는 이 같은 내용의 허위사실을 유포한 56세 여성 나 모씨를 공직선거법 위반 혐의로 구속했다고 17일 밝혔다
2. 이 소식을 접한 누리꾼들은 "남에게 상처 입히지 말라", "표현의 자유라고 하기에는 민망하다, 자유가 아니라 방종", "뭘 믿고 그런 글을 올렸을까?", "법적 처벌은 받을 것 같지만 구속할만한 범죄인지는 의문" 등의 반응을 보였다
3. 검찰에 따르면 나 씨는 지난해 대선 기간 자신의 트위터에 "박근혜의 숨겨진 아들이 은지원이고 아버지는 최태민 목사"라는 내용의 허위사실을 10여차례 올린 혐의를 받고 있다

구글 보안 책임자 "비밀번호, 태생적 한계"
2013-05-17 09:27:28
1. 그럼에도 불구하고 타브리즈는 "비밀번호를 쓰는 것 만으로 안전하다고 보기는 힘들다"며 "이 체제가 컴퓨터 보안의 핵심은 아니다"라고 지적했다
2. 그나마 이를 안전하게 보호하기 위한 방안으로 구글의 두 보안책임자는 세 가지를 제시했다
3. 타브리즈는 "불행하게도 인간은 종종 보안에서 가장 약한 연결고리"라고 말했다

야후, 뉴스피드에 유명인사·언론사 트윗 중계
2013-05-17 09:27:27
1. 야후가 제공하는 현재 뉴스와 관련 있는 정치인, 연예인, 언론사의 트윗만 뉴스피드 오른쪽에 보여주는 방식이다
2. 야후는 자사 홈페이지 뉴스피드에 실시간 트위터 중계를 하게 됐다고 16일(현지시간) 밝혔다
3. 야후는 지난 3월 뉴스 요약 앱 '섬리'를 사들이기도 했다

Figure 2-5: Korean News Summarization using Fuzzy Logic

1. 복수의 게임 관련 외신은 크라이텍이 넥슨을 통해 워페이스 서비스 지역을 일본까지 확대한다고 16일(현지시각) 보도했다
2. 그는 이어 "파트너 넥슨을 통해 진행된 한국 지역 테스트 결과에 만족한다"면서 "일본에서도 워페이스의 성공을 확신한다"고 덧붙였다
3. 체밧 옐리 크라이텍 최고경영자(CEO)는 "워페이스 일본 서비스는 우리에게 매우 중요한 발걸음"이라고 말했다

구글파트너 '도발'...안드로이드글래스 내놨다
2013-05-17 13:35:27

1. 기능은 구글글래스와 유사하고 안드로이드OS를 사용한다

2. 구글은 지난 해에도 레콘을 I/O에 초청했었다

3. 톰 파울러 CMO는 구글글래스와 유사한 점이 있다

키 작은 내 여친 “처음엔 부럽더니...슬픈 반전”
2013-05-17 12:33:34

1. 그러나 영상 마지막에는 깜짝 놀랄 반전이 숨었다

2. 이를 본 누리꾼들은 “키 작은 여친 처음에는 부럽더니 나중에는 동정이 간다”, “왠지 남일 같지가 않다”, “뒷모습이 더 안쓰러워 보인다” 등의 반응을 보였다

3. 해외 유머 커뮤니티를 통해 국내 알려진 해당 동영상은 키 차이가 많이나는 한 남녀 커플의 두 발을 연상시키는 장면으로 시작한다

고대 5.18 사진전 훼손...일베 인증샷 논란
2013-05-17 09:27:29

1. 해당 학생회는 페이스북을 통해 1980년 당시 광주에서 계엄군이 광주 시민을 폭력적으로 진압하는 사진 30여장을 전시했으며 이중 일부가 훼손됐다고 17일 밝혔다

2. 학생회 측은 기존 전시물 위에 광주민주화운동이 북한에 의해 일어난 폭동이라는 주장을 담은 사진 10여장이 붙어있었다고 설명했다

3. 사진출처 : 고려대 문과대 페이스북이와 관련해 보수 성향 인터넷커뮤니티 ‘일간베스트 저장소(일베)’의 한 게시판에는 ‘좌빨천국 고려대학교 산업화 시전’이라는 제목으로 해당 사진전을 훼손한 인증 사진과 글이 올라왔던 것으로 알려졌다

앨 고어, 트위터 공동창업자 새 사업에 투자
2013-05-17 09:27:29

1. 보도에 따르면 비즈 스톤의 오랜 동료이자 트위터를 공동 창업한 잭 도로시, 에반 윌리엄스, 리드 호프만 링크드인 CEO 외에도 세계적인 록스타 U2의 리더인 보노(Bono), 인기 미드 ‘하우스’, ‘로스트’ 제작자 그렉 야이타네스(Greg Yaitanes), 아프간 여성 사업가 로야 마붑(Roya Mahboob) 등이 젤리에 투자했다

2. (출처-젤리)앨 고어 외에도 엔젤 투자자들의 면면은 흥미롭다

3. 영화 ‘불편한 진실’로 유명한 앨 고어 미국 전 부통령이 스타트업 기업 ‘젤리’에 초기 투자를 단행했다고 씨넷이 16일(현지시간) 보도했다

“은지원이 박근혜 아들” 허위사실 유포 50대 구속
2013-05-17 09:27:28

1. 서울중앙지검 공안1부(부장검사 최성남)는 이 같은 내용의 허위사실을 유포한 56세 여성 나 모씨를 공직선거법 위반 혐의로 구속했다고 17일 밝혔다
2. 이 소식을 접한 누리꾼들은 “남에게 상처 입히지 말라”, “표현의 자유라고 하기에는 민망하다, 자유가 아니라 방종”, “뭘 믿고 그런 글을 올렸을까?”, “법적 처벌은 받을 것 같지만 구속할만한 범죄인지는 의문” 등의 반응을 보였다
3. 검찰에 따르면 나 씨는 지난해 대선 기간 자신의 트위터에 “박근혜의 숨겨진 아들이 은 지원이고 아버지는 최태민 목사”라는 내용의 허위 사실을 10여차례 올린 혐의를 받고 있다

구글 보안 책임자 “비밀번호, 태생적 한계”
2013-05-17 09:27:28

1. 그럼에도 불구하고 타브리즈는 “비밀번호를 쓰는 것 만으로 안전하다고 보기는 힘들다”며 “이 체제가 컴퓨터 보안의 핵심은 아니다”라고 지적했다
2. 그나마 이를 안전하게 보호하기 위한 방안으로 구글의 두 보안책임자는 세 가지를 제시했다
3. 타브리즈는 “불행하게도 인간은 종종 보안에서 가장 약한 연결고리”라고 말했다

야후, 뉴스피드에 유명인사·언론사 트윗 중계
2013-05-17 09:27:27

1. 야후가 제공하는 현재 뉴스와 관련 있는 정치인, 연예인, 언론사의 트윗만 뉴스피드 오른쪽에 보여주는 방식이다
2. 야후는 자사 홈페이지 뉴스피드에 실시간 트위터 중계를 하게 됐다고 16일(현지시간) 밝혔다
3. 야후는 지난 3월 뉴스 요약 앱 ‘섬리’를 사들이기도 했다

美 의회 “구글글래스 사생활 보호 대책은?”
2013-05-17 09:27:27

1. 구글 글래스에 대한 사생활 보호 문제에 대해 미국 의회가 공개 서한을 보냈다
2. 구글 글래스가 아직 출시 전임에도 불구하고 이에 대한 사회 전반적인 우려는 상당한 것으로 보인다
3. 미국 의회가 구글 글래스에 대한 개인정보보호 문제에 우려를 표시했다

3D프린팅, 21세기 대중의 연금술
2013-05-17 08:56:38

1. 이들 눈에 연금술은 손을 쓰는 천한 일에 불과했고 종교에 반하는 신비주의적 요소도 다분했다

2. 특정 사업의 기반이 3D프린팅으로 위협당할 가능성이 높아질수록 그에 대한 규제의 목소리도 커질 수밖에 없다

3. 이에 미국처럼 3D프린팅을 통해 어떤 물건을 만드는 시도나 그 설계도를 온라인으로 공유하는 행위를 법으로 금지하려는 움직임이 앞으로도 계속될 수 있다

네이버 지도 앱, 2천만 다운로드 돌파
2013-05-17 07:54:57

1. NHN은 자사 '네이버 지도'의 모바일 앱이 2천만 다운로드를 돌파, 오는 29일까지 기념 이벤트를 개최한다고 밝혔다

2. 응모 방법은 지도 앱 최신 버전을 내려 받거나 업데이트 한 뒤 실행해 '20,000,000 NAVER MAP' 중 한 글자를 찾으면 된다

3. 이벤트 당첨자는 내달 10일 개별 메일과 네이버 지도 블로그를 통해 발표된다

루게릭 · 파킨슨 발병원인 밝혀낸 국내 과학자
2013-05-17 07:24:11

1. 국내 과학자가 세계 최초로 루게릭, 파킨슨병 등 퇴행성 뇌질환 발병 원인을 밝혀냈다

2. 임정훈 교수는 "어택신2의 새로운 기능은 관련 퇴행성 뇌질환 환자들에게서 나타나는 수면 질환을 설명할 뿐만 아니라 앞으로 어택신2에 의한 퇴행성 뇌질환 발병 원인 연구와 치료에 새 패러다임을 제시할 것"이라고 전했다

3. 그동안 퇴행성 뇌질환 발병의 주요 요소인 어택신2 유전자의 분자 생물학적 기능, 특히 퇴행성 뇌질환을 일으키는 신경 세포학적 역할에 대해서는 규명된 바가 없었다

'여성운동계 대모' 박영숙 전 안철수재단 이사장 별세
2013-05-17 06:52:57

1. 지난해 2월부터 안철수재단(현 동그라미 재단) 이사장으로 활동을 이어왔다

2. 여성운동계 원로인 박영숙 전 안철수재단 이사장이 향년 81세로 17일 오전 별세했다

3. 박 전 이사장은 13대 국회의원 선거에서 전국구 1번으로 정계 입문했으며, 평민당 총재 권한대행, 민주당 최고위원 등을 지냈다

인턴 父 "윤창중 2차 성추행.." 누리꾼 분노
2013-05-17 06:52:57

1. 이에 따라 윤 전 대변인의 성추행 의혹 수사가 경범죄를 넘어 중범죄 혐의로 수사 확대될 가능성이 제기됐다

2. 이른바 '윤창중 사태'가 장기화되고 있는 가운데 파장이 좀처럼 가라앉을 줄 모른다

3. "(W호텔에서) 허락없이 엉덩이를 움켜쥔" 1차 성추행보다 윤 전 대변인이 숙소인 페어팩스호텔에 와서 또 다시 성범죄를 시도했기 때문에 경찰에 신고하게 된 것이라는 얘기다

복권 4번 당첨된 남성 '미스터 럭키' 누구?
2013-05-17 06:52:57

1. 남성은 비결을 묻는 질문에도 "그저 적절한 장소에서 정확한 시간에 복권을 샀을 뿐"이라고 말한 것으로 전해졌다

2. 한편 미국서 복권 4번 당첨된 남성이 나온 것은 이번이 처음은 아니다

3. 보도에 따르면 이 남성은 최근 이른바 '긁는 복권'에 당첨돼 2등 금액인 50만달러(약 5억6천만원)를 수령했다

페이스북 IPO 1주년…주가 30% 폭락
2013-05-17 06:22:19

1. 16일(현지시간) 월스트리트저널(WSJ) 등 주요 외신들은 이날 뉴욕증시 나스닥에서 페이스북이 전날보다 1

2. 상장 이후 페이스북에게 최대 위기가 찾아온 시기는 작년 9월이었다

3. 9월 4일 페이스북 주가는 뉴욕증시 나스닥에서 주당 17

람보르기니의 위용
2013-05-17 06:22:19

1. ¡사진=씨넷¿람보르기니 탄생 50주년을 맞아 페막 갈라쇼에서 공개된 람보르기니 에고이스타

2. 람보르기니사가 배트맨에게 어울릴 만한, 배트맨 우주전투기로 착각하게 할 만한 멋진 컨셉트카를 공개했다

3. 5200cc짜리 V10엔진으로 가는 람보르기니 에고이스타는 600마력의 힘을 자랑한다

이외수 트윗글 무단 복제 출판사 벌금
2013-05-17 06:22:19

1. 서울남부지법 형사10단독 이의진 판사는 소설가 이외수씨의 트위터 글을 무단 복제·배포한 혐의(저작권법위반)로 기소된 A 출판사와 이 회사 대표 김모(51)씨에게 벌금 1천500만원을 선고했다고 16일 밝혔다

2. 김씨는 작년 2~5월 이씨가 자신의 트위터에 올린 “변명을 많이 할수록 발전은 느려지고 반성을 많이 할수록 발전은 빨라진다” 등의 글 56개를 무단으로 복제해 ‘이외수 어록 24억짜리 언어의 연금술’이라는 제목의 전자책 파일로 만들어 배포한 혐의로 기소됐다

3. 이 판사는 “이씨의 트윗글은 짧아도 그 속에 삶의 본질을 꿰뚫는 촌철살인의 표현과 시대와 현실을 풍자하는 독창적인 표현형식이 대부분 포함돼 있어 이씨의 개성을 드러내기에 충분하다”며 “무단 복제된 글들은 이씨의 사상 또는 감정이 표현된 글로서 저작물로 봐야 한다”고 판시했다

303억 다이아몬드 “왜 그렇게 비싼거야?”
2013-05-17 05:51:32

1. 73캐럿으로 지금까지 경매에 붙여진 다이아몬드 중 가장 큰 크기를 자랑한다

2. 다이아몬드를 낙찰 받은 곳은 보석 및 시계 거래 회사인 해리 윈스턴이며, 이들은 이 다이아몬드에 위스턴 레거시라는 이름을 붙였다

3. 해외 주요 매체에 따르면 15일(현지시각) 스위스 제네바에서 열린 크리스티 보석 경매에서 보츠와나산 다이아몬드가 2천670만달러(한화 약 303억원)에 낙찰됐다

2.7 Conclusions

The system is so simple, but the result is very good. In the internet community(clien), many people have agreed that this summarization result is great. We feel that even the simple rule(technic) can make our lives a lot simpler and fluent. So our scientist’s work directly influences human life and we feel very good about the progress of the work we are doing. We’re proud of ourselves.

For the system evaluation, there is not any Korean text summarization data set, including commercial software. So we didn’t compare this result. However, subjectively this is not so obsolete. If we add more other Intelligent systems or algorithms, this result can be much greater.

2.8 Further Work

There are various kind of summarization; genetic algorithm[13], Neural Network[14], Cluster based method, tf-idf method and so on. I’m particularly interested in the graph theoretic approach. The small world concept is well-known in biologic, social network and statistical physics. In 2001, the ‘KeyWorld’ paper showed that the small world structure also exists in documents.[15] Based on the topology, they developed an indexing system called *KeyWorld*, which extracts important terms by measuring their contribution to the graph being small world.

We can improve it, current korean summarization system, using this graph topology, and add the network measures as well as feature measurements. This will be funny further work.

3

Korean Text Clustering System and Method

In last chapter, we talked about the korean text summarization. However it has a big problem. A lot of news articles is publishing a day. In the case of South korea, $30,000 \sim 50,000$ news articles are published each day. The summarization works for a one article. It means the amount of summarized sentences increases with the count of articles. As you have seen, we can easily access a lot of news articles today as if we really doesn't know. So we need to get more convenient tool to solve it. That is a clustering method.

Clustering is an unsupervised learning technique that has been widely used in the process for finding subgroups, or clusters, in a data set. Typically, a common form of text processing in many information retrieval systems is based on the analysis of word occurrences across a document collection. The number of words/terms used by the system defines the dimension of a vector space in which the analysis carried out. Reduction of the dimension may lead to significant savings of computer resources and processing time. However, poor feature selection may degrade the system performance. [16]

In this context, our research focuses on modeling using language feature, especially Korean one of Asian languages. The modern English alphabet consisted of 26 letters and word is constructed by listing of letters. In the other hands, one letter of the Asian alphabet has more meaningful. For example, Chinese '광'(*gwang*) is translated into 'light'. That is, one letter has more information. Due to this characteristic, we select the syllable feature of Asian language for dimension reduction.

3.1 Syllable vector

We propose a fast document clustering method based on the language property, especially Korean documents. Korean language has a property that one syllable has more information than

English alphabet. Thus we construct the syllable vector of each document. Our experimental evaluations show that the proposed document clustering method is efficient for speed and effective for selecting candidates.

In the conventional methods, in order to calculate information of text they consider the granularities such as terms, sentences, paragraphs or documents. But Asian language has a different birth background. Additionally we consider the syllable-based calculation.

The Korean alphabet, known as Hangul in South Korea, is divided into three parts: ch'osong(initial consonant), chungsong(peak vowel) and chongsong(final consonant). This is a basic framework that King Sejong and the Chiphyonjon scholars adhered to when creating the letters. Chongsong was not separately created and was a repetition of the ch'osong. Therefore, Hangul letters are block with consonants and vowels.[17]

Each syllabic block consists of two to six letters, including at least one consonant and one vowel. Each Korean word consists of one or more syllables, hence one or more blocks. The number of mathematically possible distinct blocks is $11,172$, 가 - 힣.

$$\mathrm{Dim}(\{\text{가}, \text{각}, \text{갂}, \text{갃}, \cdots, \text{힣}\}) = 11,172 \tag{3.1.1}$$

Typically, the vector space model of document has a big dimension for words or terms. However through the syllable block size, we can reduce it into $11,172$ dimension for Asian document. Besides statistically the number of frequent syllables is about $1,200$. Without loss of generality, the dimension of Korean document can be $1,200$.

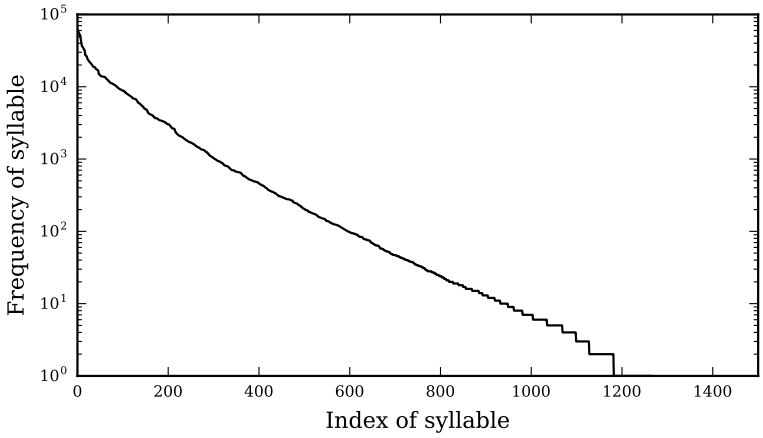


Figure 3-1: Frequency of syllables

3.1.1 Vector space model

Vector space model has been widely studied for a long time [20]. It can be explained by constructing of matrix from documents. Let's define basic components [21].

A corpus is a pair $\mathcal{C} = (\mathcal{T}, \mathcal{D})$, where $\mathcal{T} = \{\boldsymbol{t}_1, \cdots, \boldsymbol{t}_m\}$ is a finite set whose elements are referred to as terms, and $\mathcal{D} = (\boldsymbol{d}_1, \cdots, \boldsymbol{d}_n)$ is a set of documents. Each documents $\boldsymbol{d}_i$ is a finite sequence of terms, $\boldsymbol{d}_j = (t_{j1}, \cdots, t_{jk}, \cdots, t_{jl_j})$. Let $\mathcal{C}$ be a corpus such that $|\mathcal{T}| = m$ terms and $\mathcal{D}$ contains n documents. If t_i is a term and $\boldsymbol{d}_j$ is a document of $\mathcal{C}$, the frequency of t_i in $\boldsymbol{d}_j$ is the number of occurrences of t_i in $\boldsymbol{d}_j$, that is,

$$a_{ij} = |\{p \mid t_{jp} = t_i\}| \tag{3.1.2}$$

where p is a position of the term. The frequency matrix of the corpus $\mathcal{C}$ is the matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ defined by $A = (a_{ij})$. It is called **term-document matrix** or TF(Term Frequency) matrix. Each term t_j generates a row vector $(a_{i1}, a_{i2}, \cdots, a_{in})$ referred to as a term vector and each document $\boldsymbol{d}_j$ generates a column vector

$$\boldsymbol{d}_j = \begin{bmatrix} a_{1j} \\ \cdots \\ a_{mj} \end{bmatrix} \tag{3.1.3}$$

For example, there are three documents; $\boldsymbol{d}_1 = \{\text{apple}, \text{banana}, \text{kiwi}\}$, $\boldsymbol{d}_2 = \{\text{apple}, \text{banana}, \text{store}\}$, $\boldsymbol{d}_3 = \{\text{store}\}$. It will be the following matrix $\boldsymbol{A}$;

$\boldsymbol{A} =$

	$\boldsymbol{d}_1$	$\boldsymbol{d}_2$	$\boldsymbol{d}_3$
apple	1	1	0
banana	1	1	0
kiwi	1	0	0
store	0	1	1

Table 3-1: Example of term-document matrix $\boldsymbol{A}$

As we can see, a length of rows is dependent on the count of distinct terms. However, the proposed method can solve it as to reduce the dimension of the matrix.

3.1.2 Definition of Syllable Vector

A document is consist of sentences, a sentence is consist of words and a word is consist of syllables. To make a effective term-document matrix, we use a syllable feature of asian text data. A syllable is a unit of organization for a sequence of speech sounds. In english, the word *math* is composed of two syllables: *ma* and *th*. But the letters of Korean are grouped into blocks, such as 수 *su*, each of which transcribes a syllable. That is, although the syllable 수 *su* may

look like a single character, it is actually composed of two letters: ㅅ *s* and ㅜ *u*. The syllable of Korean has more information, although the length of syllable is only one.

Assume w_i is some observable word which is consist of syllables n.

$$w_i = s_{i1}s_{i2}s_{i3}\cdots s_{il_i} \tag{3.1.4}$$

Syllable vector is composed of a syllable as a term. For example, *수학 suhak* [1] is divided into two terms: *수 su* and *학 hak*. Then how we make Syllable vector? We propose two types approaches.

3.1.2.1 Syllable-n Vector

Given a data w_i, we can select only a first syllable. In Korean, the first syllable has more information than other position syllables. The reason is that many of korean words transliterated using the Chinese character. For example, a one syllable *한 han* of *한국 hankuk* indicate Korea which is transliterated by the Chinese character *韓*. Or *선생 seunsaeng* of *선생님 seunsaengnym* has all information as a *teacher* except for polite feeling. In the realistic articles, a syllable *홍* has a most important information among the words *홍길동 선생님*, *홍 선생님*, *홍길동*. Therefore we suppose three types Syllable-n vector.

Algorithm 1 Syllable-n vector

Input: A document $\boldsymbol{d}_i$
Output: Syllable-n set, $\boldsymbol{s}_j = (t_{j1}, \cdots, t_{jk}, \cdots, t_{jl_j})$
1: Extract words from a given document.
2: Decompose the word w_j into the syllable set $\boldsymbol{w}_j = \{s_{j1}, s_{j2}, \cdots, s_{jl_j}\}$
3: **loop** Consider n syllables from the syllable set $\boldsymbol{w}_j$ of w_j
4: $t_{jp} = s_{j1}\cdots s_n$
5: **end loop**

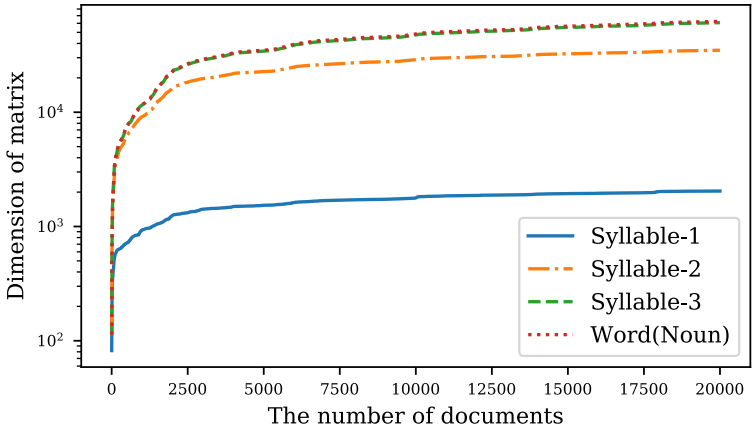


Figure 3-2: Dimension reduction using Syllables-n vector

[1] *수학* is a word 'math' in English.

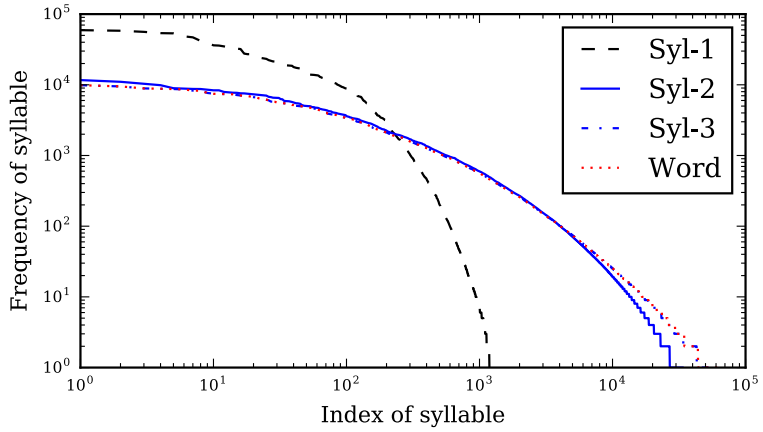


Figure 3-3: Sorting plot of syllable by frequency

Type	Syl-1	Syl-2	Syl-3	Word
Top rank	기 62399	말 13004	말 13004	말 13004
	대 59288	한국 11661	경우 9875	경우 9875
	사 58175	서울 11029	한국 9484	한국 9484
	이 56088	지난 10355	서울 8892	서울 8892
	전 53797	경우 9875	업체 8828	업체 8828
	지 53451	기업 8947	시장 8711	시장 8711
	수 48514	업체 8828	지난해 8605	지난해 8605
	경 46920	투자 8741	기업 8277	기업 8277
	시 41783	시장 8719	조사 8251	조사 8251
	주 39078	자동 8549	수출 7672	수출 7672
	부 36179	수출 8358	자동차 7483	자동차 7460
	자 35890	조사 8353	정부 7425	정부 7425
	정 34936	은행 7929	지역 7351	지역 7351
	공 33698	사업 7818	개발 7346	개발 7346
	가 33482	관계 7729	은행 7338	은행 7338
Total Count	1,266	33,639	59,574	61,176

Table 3-2: Top rank syllables for n=1, 2, 3 and words

3.1.2.2 Syllable-n-All Vector

Our proposed methods have two types. Syllable-n vector has a forehead information of a word. But the loss of information can be larger than the benefit for dimension reduction. Then our second method is Syllable-n-All vector. This method use a combination of syllables.

Algorithm 2 Syllable-n-All vector

Input: A document $\boldsymbol{d}_i$
Output: Syllable-n-All set, $\boldsymbol{s}_j = (t_{j1}, \cdots, t_{jk}, \cdots, t_{jl_j})$
1: Extract words from a given document.
2: Decompose the word w_j into the syllable set $\boldsymbol{w}_j = \{s_{j1}, s_{j2}, \cdots, s_{jl_j}\}$
3: **loop** Consider n syllables from the syllable set $\boldsymbol{w}_j$ of w_j
4: Set the position pairs as much as $\boldsymbol{c} = \binom{l_j}{n}$
5: $t_{jp} = \{s_{j\boldsymbol{c}_1} \cdots s_{j\boldsymbol{c}_q}, \cdots, s_{j\boldsymbol{c}_{q-1}} \cdots s_{j\boldsymbol{c}_q}\}$
6: **end loop**

3.2 Measurement of similarity

Computation of text similarity is a fundamental problem in information retrieval. [22] With various data features, there exists a lot of measurement like Cosine similarity, Kullback-Leibler Divergence, Kendall's τ coefficient, Binary Distance Measures, etc.

To show the information loss of Syllable vector comparing to conventional word vector, we use Cosine similarity and Pearson Correlation.

3.2.1 Cosine Similarity

Given two vectors, $\boldsymbol{a}$ and $\boldsymbol{b}$, the cosine similarity, $\cos(\theta)$, is represented using a dot product and norm as

$$\cos(\theta) = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\|\boldsymbol{a}\|_2 \|\boldsymbol{b}\|_2} = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\sqrt{\boldsymbol{a} \cdot \boldsymbol{a}}\sqrt{\boldsymbol{b} \cdot \boldsymbol{b}}} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}}, \tag{3.2.1}$$

where a_i and b_i are components of vector $\boldsymbol{a}$ and $\boldsymbol{b}$ respectively.

The range of similarity values is $[-1, 1]$. The value 1 means exactly the same(similar), -1 is exactly the opposite(dissimilar). And the value 0 indicate orthogonality (decorrelation).

Let similarity be a Euclidean distance of vectors. When we measure the distance between two vectors in Euclidean space, we use a Euclidean norm (L^2 norm).

$$d(\boldsymbol{a}, \boldsymbol{b}) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_n - b_n)^2} = ||\boldsymbol{a} - \boldsymbol{b}|| \tag{3.2.2}$$

It can be observed like that

$$\|\boldsymbol{a}-\boldsymbol{b}\|^2 = (\boldsymbol{a}-\boldsymbol{b})^\top(\boldsymbol{a}-\boldsymbol{b}) = \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 - 2\boldsymbol{a}^\top\boldsymbol{b} \tag{3.2.3}$$

When $\boldsymbol{a}$ and $\boldsymbol{b}$ are normalized to unit length, that is, $\|\boldsymbol{a}\|^2 = \|\boldsymbol{b}\|^2 = 1$, the similarity is that

$$\text{similarity} = d(\boldsymbol{a},\boldsymbol{b}) = \sqrt{2(1-\cos(\boldsymbol{a},\boldsymbol{b}))} \tag{3.2.4}$$

Thus the similarity will range from 0 meaning exactly the same to 2 meaning exactly opposite. In our experiments, the application[2] use it for calculating nearest neighbors points. (See Figure 3-6 $\sim$ 3-11)

3.2.2 Pearson Correlation

Pearson correlation is a measure of the extend to which two vectors are related.[23, 24] A common used form is

$$\rho_{\boldsymbol{a},\boldsymbol{b}} = \frac{\operatorname{cov}(\boldsymbol{a},\boldsymbol{b})}{\sigma_{\boldsymbol{a}}\sigma_{\boldsymbol{b}}} = \frac{\mathrm{E}[\boldsymbol{a}\boldsymbol{b}] - \mathrm{E}[\boldsymbol{a}]\,\mathrm{E}[\boldsymbol{b}]}{\sqrt{\mathrm{E}[\boldsymbol{a}^2] - [\mathrm{E}[\boldsymbol{b}]]^2}\,\sqrt{\mathrm{E}[\boldsymbol{b}^2] - [\mathrm{E}[\boldsymbol{b}]]^2}} \tag{3.2.5}$$

where cov is a covariance, $\sigma_{\boldsymbol{a}}$, $\sigma_{\boldsymbol{b}}$ is a standard deviation of $\boldsymbol{a}$, $\boldsymbol{b}$, E is a expectation.

Its range is $[-1, 1]$ and it is 1 when $\boldsymbol{a} = \boldsymbol{b}$.

3.2.3 Empirical analysis and results

In order to evaluate the proposed method, we conduct the experiements under the HKIB-20000 benchmark collections for text categorization research. The HKIB-20000 test collection is a modified version of the HKIB-40075 data set where 20,000 documents were carefully selected from 40,075 documents of HKIB-40075 [19].

Following text (Figure 3-4) is a sample of HKIB-20000. It has pre-defined data: DocID, Category03, Category07. We use it to compute F1-measure.

For HKIB-20000, the count of total articles is 20,000. First, we construct a term-frequency matrix. Stop words are disregarded. We only use noun words. Then the matrix size is a $(61,176 \times 20,000)$. It is a huge number as a matrix. And every document vector is a sparsity. On average the number of non-zero elements are 86(0.001%) for every articles (Fig 3-5). Similarly we make matrices for Syllable-1, Syllable-2, Syllable-3, Syllable-1-All and Syllable-2-All.

[2] Python library ANNOY(/spotify/annoy, Approximate Nearest Neighbors Oh Yeah)

#DocID : 10646
#CAT'03: /건강과 의학/의약학/한의학 전통의학
#CAT'07: /정치/정부부처/보건복지부
#TITLE : 한약재 규격품사용 의무화
#TEXT :
보사부는 불량 한약재 유통을 막기 위해 내년 4월이후 한의원 등
한방취급의료기관에 포장과 성분을 통일한 규격 한약재 사용을
의무화하기로 했다.

보사부는 30일 대한약전과 생약규격집에 실린 국내 한약재 5백14종 가운
데 사용빈도가 높은 녹용,우황,당귀,작약 등 모두 37개 품목에 대해 규격
품만을 사용하도록지정 고시했다.

이들 한약재는 내년 4월부터 KGMP(우수품질관리기준)시설을 갖춘 전문제
약업소만이 생산,유통시킬수 있으며 한의원,약국 등 한방취급 의료기관은
이 규격품만을사용해야 하며 비규격품 사용은 금지된다.

또 이들 전문제약업소는 한약재의 세척-건조-절단-포장 과정에서 엄격한
품질검사를 실시해야 하며 반드시 원산지 표시를 해야 한다.

보사부는 이밖에 규격품목의 범위를 점진적으로 넓혀나가고 또 포장단위
는 갈근,감초 등 일반 한약재의 경우 6백g, 녹용,우황 등 고가 한약재는
10g으로 제한하도록 할 방침이다.

이번 조치는 유통중인 대다수 한약재가 농가에서 재배된 뒤 건조, 절단,
정제후시중에 그대로 유통되면서 품질검사를 거치지 않아 품질이 낮고 또
일반 농산물과한약재의 구분이 명확하지 않아 가격이 불안정하거나 거래질
서가 문란하다는 지적에따른 것이다.

대상 한약재는 갈근,감국,감초,건강,계지,계피,곽향,구기자,길경,녹각,녹
용,당귀,도인,마황,반하,복령,부자,사삼,산수유,산조인,산약,숙지황,시호
,신곡,우황,육계.작약,저령,진피,천궁,행인,향부자,황금,황기,황련,황백,
후박 등이다.

Figure 3-4: A sample document of HKIB-20000

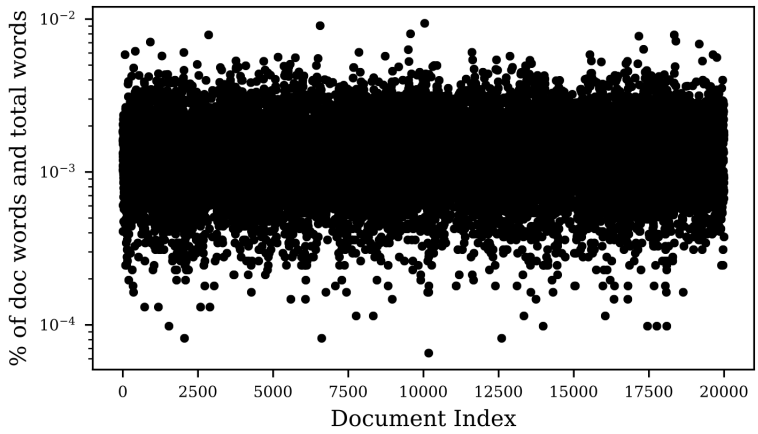


Figure 3-5: Percentage of document words and total words

Without conventional methods for matrix like TF-IDF, LSI, we only compare with term-frequency vectors. For example, Figure 3-6 is a cosine similarity plot of document number 5222. X-axis means a every index of documents($\sim 20,000$), Y-axis is a value of cosine similarity with document number 5222.

	Syllable-1	Syllable-2	Syllable-3	Syl-1-All	Syl-2-All	Syl-3-All	Words
Dimension	1,266	33,639	59,574	1,379	57,541	91,528	61,176
Memory	194M	5.1G	8.9G	211M	8.6G	-	9.2G

Table 3-3: Specification of each matrix for HKIB-20000

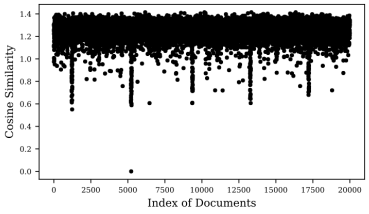


Figure 3-6: Syllable-1

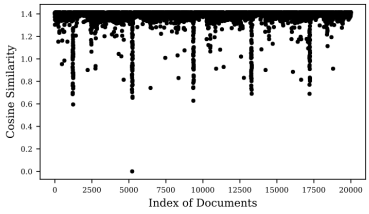


Figure 3-7: Syllable-2

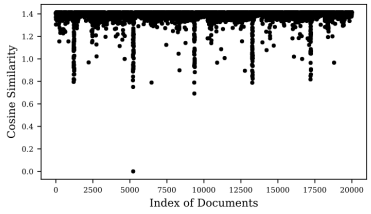


Figure 3-8: Syllable-3

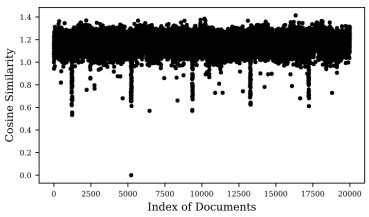


Figure 3-9: Syllable-1-All

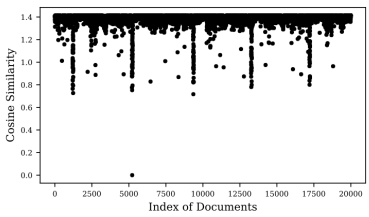


Figure 3-10: Syllable-2-All

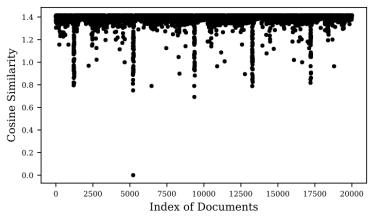


Figure 3-11: Word

To show the efficiency of the proposed method, we run Pearson correlation of cosine similarity. For one document, total consume time is 610 sec(about 10 min). More details refer Table 3-4. If we run it for all documents, we have to conduct for 138 days on the single process. So we select the 20 samples: Document number 5, 7, 10, 12, 14, 22, 43, 69, 71, 74, 78, 83, 95, 98, 100, 101, 105, 108, 110, 202.

	Syllable-1	Syllable-2	Syllable-3	Syl-1-All	Syl-2-All	Word	Total
Ave. Time	3.36	85.50	169.23	3.12	156.86	192.35	610.41
Tot. Time	67.11	1710.01	3384.60	62.40	3137.20	3846.93	12208.25

Table 3-4: Consuming time for similarity between 20 sample set and all other documents

We conduct the calculation of similarity for samples like Fig 3-6 $\sim$ 3-11 and Pearson correlation apply for all case. Fig 3-22 is a average of pearson correlation for 20 samples. The remarkable point is that **Word** row is highly similar to **Syl-2**, **Syl-2-All**.

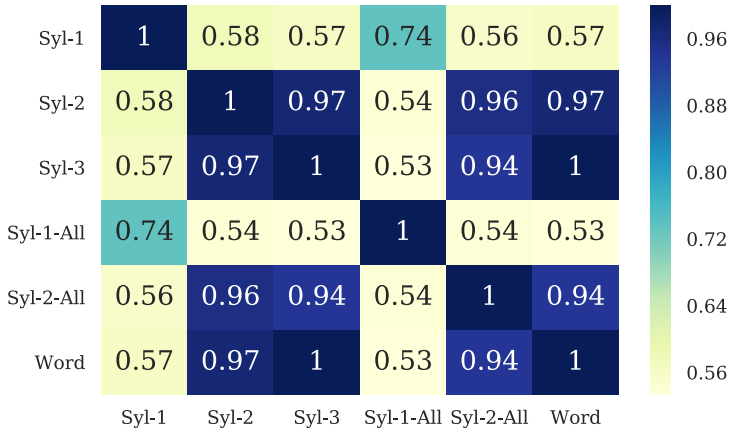


Figure 3-12: Heat map by Pearson correlation for all documents

To show the performance of the proposed method, we compare the computation time in the real-world data set that korean news for days by web-crawling(2015). When we tried for 0 - 60,000 news documents, the proposed method is well work. However traditional method is not work with memory issue(bigger than 15GB). In the Fig 3-13, the computation time of word-based matrix is exponentially increasing, but the proposed method has almost linearly increasing performance.

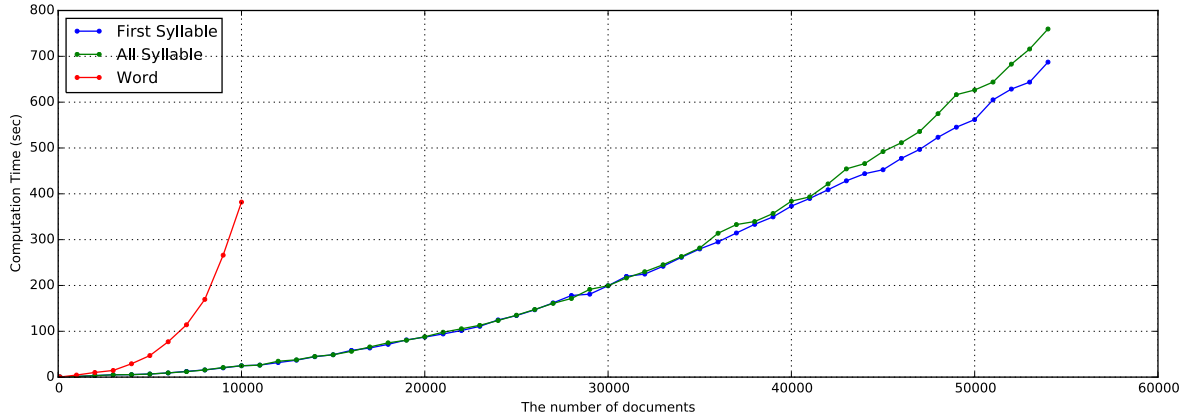


Figure 3-13: Comparison for the computation time (Word, Syllable-1, Syllable-1-All)

3.3 Latent semantic indexing

The latent semantic indexing (LSI) is a traditional method for information-retrieval, based on singular value decomposition(SVD) [26, 27]. SVD is a common technique for analysis of multivariable data. But the computation cost of matrix factorization is very expensive. In this chapter, we show that the proposed method is effective for LSI. And we talk about TF-IDF matrix, SVD, LSI(Latent Semantic Indexing) and probablistic LSI with empirical experiments.

3.3.1 TF-IDF

In previous chapter, Term Frequency(TF) matrix is used for the calculation of cosine similarity. In information retrieval(IR), the commonly used matrix is TF-IDF(Term Frequency-Inverse Document Frequency). In section 3.1.1, we defined TF matrix with the raw count of a term in a document, but normally we use a normalized term frequency. Let tf(t, $\boldsymbol{d}$) be a term frequency of term t in document $\boldsymbol{d}$. Simplify, tf(t, $\boldsymbol{d}$) = $f_{t,\boldsymbol{d}}$.

$$\text{term frequency} = f_{t,\boldsymbol{d}} / \sum_{t' \in \boldsymbol{d}} f_{t',\boldsymbol{d}} \tag{3.3.1}$$

Weighting scheme	TF weight
Raw count	$f_{t,\boldsymbol{d}}$
Logarithm	$1 + \log(f_{t,\boldsymbol{d}})$
Augmented	$0.5 + 0.5 \cdot \frac{f_{t,\boldsymbol{d}}}{\max_{t' \in \boldsymbol{d}}(f_{t',\boldsymbol{d}})}$
Boolean	0, 1
Log ave.	$\frac{1+\log(f_{t,\boldsymbol{d}})}{1+log(\text{ave}_{t \in \boldsymbol{d}}(f_{t,\boldsymbol{d}})}$

Table 3-5: Variants of term frequency(TF) weight

Now define the inverse document frequency(IDF). The goal of document frequency is to scale down the impact of terms that occur very frequently in a given corpus and that are hence empirically less informative than feature that occur in a small fraction of the corpus. For example, the terms *the*, *a*, *in* are frequent at the most of documents. These are not important terms of document and not be helpful for information retrieval.

$$\text{idf}(t, \boldsymbol{D}) = \log \frac{N}{|\{\boldsymbol{d} \in \boldsymbol{D} : t \in \boldsymbol{d}\}|} \tag{3.3.2}$$

with N is a total number of documents in the corpus $N = |\boldsymbol{D}|$. In the industrial fields[3], to prevent zero divisions, the constant 1 is added to the numerator and denominator of the idf,

Then TF-IDF is calculated as

$$\begin{aligned} \text{TF-IDF}(t, \boldsymbol{d}, \boldsymbol{D}) &= \text{tf}(t, \boldsymbol{d}) \cdot \text{idf}(t, \boldsymbol{D}) \\ &= \left(f_{t,\boldsymbol{d}} / \sum_{t' \in \boldsymbol{d}} f_{t',\boldsymbol{d}} \right) \cdot \left(\log \frac{1+N}{1+|\{\boldsymbol{d} \in \boldsymbol{D} : t \in \boldsymbol{d}\}|} + 1 \right) \end{aligned} \tag{3.3.3}$$

3.3.2 SVD

The SVD(singular value decomposition) is a factorization of a real or complex matrix. It is a useful starting point in many algorithms and it has been described as the "Swiss Army knife of matrix decompositions" [30, 21]. The SVD is motivated by the geometric fact:

> The image of the unit sphere under any $m \times n$ matrix is a hyperellipse. [25]

Let m and n be arbitrary. Given $\boldsymbol{A} \in \mathbb{C}^{m \times n}$, not necessarily of full rank, the SVD of $\boldsymbol{A}$ is a factorization

$$\boldsymbol{A} = \boldsymbol{U} \Sigma \boldsymbol{V}^T \tag{3.3.4}$$

where $\boldsymbol{U} \in \mathbb{C}^{m \times m}$, $\boldsymbol{V} \in \mathbb{C}^{n \times n}$ are unitary matrices, $\Sigma \in \mathbb{R}^{m \times n}$ is a diagonal matrix.

[3] TfidfTransformer of PYTHON library scikit-learn

For example, given any matrix can be decompostion like Figure 3-14.

	d1	d2	d3
w1	1	0	0
w2	0	1	0
w3	1	1	1
w4	1	1	0
w5	0	0	1

=

-0.27	0.21	0.70	-0.53	0.30
-0.27	0.21	-0.70	-0.53	0.30
-0.71	-0.33	0	-0.10	-0.60
-0.55	0.43	0	0.64	0.29
-0.15	-0.77	0	0.10	0.60

2.35	0	0
0	1.19	0
0	0	1.00
0	0	0
0	0	0

-0.65	0.26	0.70
-0.65	0.26	-0.70
-0.36	-0.92	0

Figure 3-14: Example for singular value decomposition

The computation of the SVD of the matrix $\boldsymbol{A}$ is related to the eigenvalue decompositino of the matrix $\boldsymbol{A}^T\boldsymbol{A}$. Basically, we might calculate the SVD of $\boldsymbol{A}$ as follows.

Algorithm 3 Computating the SVD using eigenvalue problem

Input: A matrix $\boldsymbol{A}$
Output: The SVD of $\boldsymbol{A}$, $\boldsymbol{U}\Sigma\boldsymbol{V}^T$

1: Form the covariance matrix of $\boldsymbol{A}$, $\boldsymbol{A}^T\boldsymbol{A}$
2: Compute the eigenvalue decomposition $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{V}\Lambda\boldsymbol{V}^T$
3: Let Σ be the $m \times n$ nennegative diagonal square root of Λ
4: Solve the system $\boldsymbol{U}\Sigma = \boldsymbol{A}\boldsymbol{V}$ for unitary matrix $\boldsymbol{U}$ using scheme such as QR factorization

However, the Algorithm 3 may be much more sensitive to perturbations, because it reduces the SVD to an eigenvalue problem. Thus usually an alternative way use a QR decomposition such as the Algorithm 4.

Algorithm 4 Computing the SVD using QR decomposition

Input: A matrix $\boldsymbol{A}$
Output: The SVD of $\boldsymbol{A}$, $\boldsymbol{U}\Sigma\boldsymbol{V}^T$

1: Compute the QR factorization of the matrix $\boldsymbol{A}$, $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R}$
2: Reduce the upper triangular matrix $\boldsymbol{R}$ to a bidiagonal matrix $\boldsymbol{B}$ using orthogonal transformation, $\boldsymbol{R} = \boldsymbol{U}_1\boldsymbol{B}\boldsymbol{V}_1$
3: Reduce the bidiagonal matrix $\boldsymbol{B}$ to a diagonal matrix Σ using an iterative method, $\boldsymbol{B} = \boldsymbol{U}_2\Sigma\boldsymbol{V}_2$
4: Calculate that $\boldsymbol{A} = (\boldsymbol{Q}\boldsymbol{U}_1\boldsymbol{U}_2)\Sigma(\boldsymbol{V}_2\boldsymbol{V}_1) = \boldsymbol{U}\Sigma\boldsymbol{V}^T$

But think about the size of TF matrix in the chapter 3.2.3. Matrix is so huge, then we abandon the accuracy of computation and get the efficiency for computation time (data size/accuracy tradeoff) using data projection method. A well-known algorithm is a Randomized SVD in the data mining fields. [31, 32] A randomized matrix algorithms are a class of recently-developed with random sampling and random projection algorithms

3.3.3 LSI

We have seen that TF-IDF is an efficient and simple scheme for matching words. However it can't capture the similar documents using different words. To solve this shortcoming, new novel method is proposed, most notably LSI(Latent Semantic Indexing) or LSA(Latent Semantic Analysis). [27]

LSI uses SVD of the matrix to identify a linear subspace in the space of term-frequency documents. The derived features of LSI, which are linear combinations of the original TF-IDF features, can capture some linguistic properties. [29]

Let's take the example from Fig 3-14. We choose the k which smaller than the rank of given matrix. Then only computes the k largest singular values which is called *low-rank approximation*. Following example show $k = 2$. After this calculation, the matrix makes the linguistic analysis like polysemy, synonym. In the original matrix(Fig 3-14), w1 and w2 are completely different terms. However, in this process, w1 and w2 has almost same values in each documents d1, d2.

	d1	d2	d3
w1	0.5	0.5	0
w2	0.5	0.5	0
w3	1	1	1
w4	1	1	0
w5	0	0	1

=

-0.27	0.21	0.70	-0.53	0.30
-0.27	0.21	-0.70	-0.53	0.30
-0.71	-0.33	0	-0.10	-0.60
-0.55	0.43	0	0.64	0.29
-0.15	-0.77	0	0.10	0.60

2.35	0	0
0	1.19	0
0	0	0
0	0	0
0	0	0

-0.65	0.26	0.70
-0.65	0.26	-0.70
-0.36	-0.92	0

Figure 3-15: Example of latent semantic indexing (Algorithm 5)

LSI captures some structure such as polysemy or synonymy in the data which is obscured when raw term overlap is used.

Algorithm 5 Construction matrix of Latent Semantic Indexing

Input: A set of documents
Output: A successive approximations $\boldsymbol{A}'$ of matrix $\boldsymbol{N}$

1: Compute term vectors of a documents
2: Construct the document matrix $\boldsymbol{A}$ by term vectors
3: Compute the normalized matrix $\boldsymbol{N}$ by a selected method like TF-IDF in $\boldsymbol{A}$
4: Compute the SVD, $(\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V})$ of $\boldsymbol{N}$
5: Adopt a rank k, which is a user-specified parameter
6: Compute a new matrix $\boldsymbol{A}_k$ as $\boldsymbol{U}_k \boldsymbol{\Sigma}_k \boldsymbol{V}_k^T$
7: Transform the matrix $\boldsymbol{A}' = \boldsymbol{A}_k \boldsymbol{V}_k$

One of the role of LSI is the dimension reduction. Given matrix A is appriximated that

$$\boldsymbol{A} \approx \boldsymbol{A}_k = \boldsymbol{U}_k \Sigma_k \boldsymbol{V}_k^T \tag{3.3.5}$$

We use the transformed matrix

$$\boldsymbol{A}' = \boldsymbol{A}_k \boldsymbol{V}_k \tag{3.3.6}$$

In other words, the size $m \times n$ of $\boldsymbol{A}_k$ will be reduced to $m \times k$ by multiplying the matrix $\boldsymbol{V}_k$.

3.3.4 Empirical analysis and results with Syllable Vector

Using HKIB-20000, we run Pearson correlation for cosine similarity which is applied LSI. We set a rank $k = 1000$. For document number 5222, we calculate the cosine similarity for all documents.

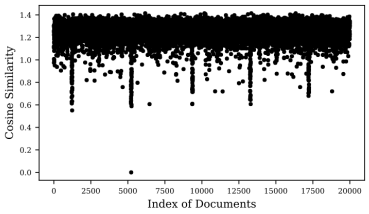


Figure 3-16: Syllable-1

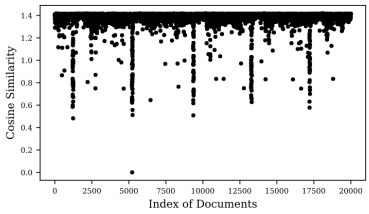


Figure 3-17: Syllable-2

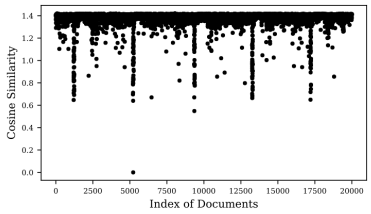


Figure 3-18: Syllable-3

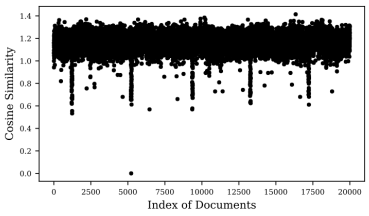


Figure 3-19: Syllable-1-All

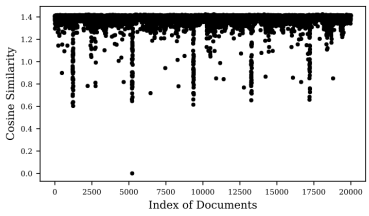


Figure 3-20: Syllable-2-All

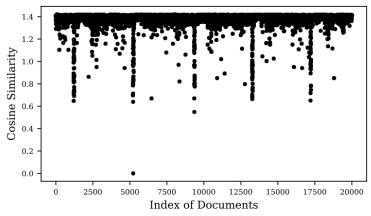


Figure 3-21: Word

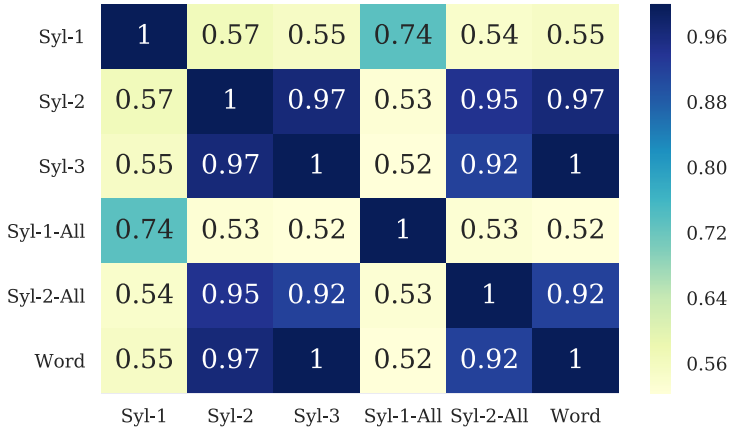


Figure 3-22: LSI heat map by Pearson correlation for all documents

	Syllable-1	Syllable-2	Syllable-3	Syl-1-All	Syl-2-All	Word	Total
Dimension	1,266	33,639	59,574	1,379	57,541	61,176	-
Memory	194M	5.1G	8.9G	211M	8.6G	9.2G	-
SVD Ave. Time	9.25	55.72	89.95	9.42	88.04	90.45	342.82
SVD Tot. Time	175.72	1058.59	1709.03	178.89	1672.84	1718.55	6513.62

Table 3-6: Consuming time for Cosine similarity between 20 sample set and all other documents

3.4 Non-negative matrix factorization

SVD is one of the most costly decompositions, both based on computation and memory consumption in the old days. In order to find the eigenvalue from a large matrix we consume a large computation time and memory. However, a new kind of factorization algorithm has emerged. The name is a non-negative matrix factorization(NMF).

3.4.1 Definition of Non-negative Matrix Factorization

NMF is a matrix factorization algorithm that finds the positive factorization of a given positive matrix [33]. Given a non-negative matrix $\boldsymbol{A}$ of size $m \times n$, where each column of $\boldsymbol{A}$ corresponds to a data point in the m-dimensional space, and a positive integer $k < \min\{m, n\}$, NMF finds two non-negative matrices $\boldsymbol{W} \in \mathbb{R}^{m \times k}$ and $\boldsymbol{H} \in \mathbb{R}^{k \times n}$ [36] so that

$$\boldsymbol{A} \approx \boldsymbol{W}\boldsymbol{H} \tag{3.4.1}$$

That is, the goal is to factorize $\boldsymbol{A}_{m \times n}$ into the nonnegative $\boldsymbol{W}_{m \times k}$ and $\boldsymbol{H}_{k \times n}$ that minimize the objective function

$$J = \frac{1}{2}||\boldsymbol{A} - \boldsymbol{W}\boldsymbol{H}|| \tag{3.4.2}$$

with the updating formulas $w_{ij} \leftarrow w_{ij} \frac{(\boldsymbol{A}\boldsymbol{H})_{ij}}{(\boldsymbol{W}\boldsymbol{H}^T\boldsymbol{H})_{ij}}$, $h_{ij} \leftarrow h_{ij} \frac{(\boldsymbol{A}^T\boldsymbol{W})_{ij}}{(\boldsymbol{H}\boldsymbol{W}^T\boldsymbol{W})_{ij}}$.

The objective function J is not increasing under the iterative updating rules and the convergence is guaranteed [34]. Note that the solution to minimizing J is not unique.

3.4.2 Non-negative Matrix Factorization with document clustering

Take the example matrix from the previous chapter (Fig 3-14). We swap the axies of the matrix $\boldsymbol{A}$ for the conventional treatments in the natural language processing(NLP) and information retrieval(IR) as $\boldsymbol{B}$.

$$B = A^T = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 1.58 & 0 \\ 1.57 & 0.01 \\ 0 & 1.42 \end{bmatrix} \times \begin{bmatrix} 0.32 & 0.32 & 0.63 & 0.63 & 0 \\ 0 & 0 & 0.70 & 0 & 0.70 \end{bmatrix}$$

The matrix $\boldsymbol{B}$ which each row means documents and each column means terms is factorized two matrices $\boldsymbol{W}$ and $\boldsymbol{H}$. The common number k of two matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ is defined as a number of features. The row of $\boldsymbol{W}$ is a index for each document, then each values of one row means the weight of each feature group. In our example, the document 1 has 1.58 wieght for the first feature, and 0 weight for the second feature. Through the matrix $\boldsymbol{W}$, we know the document 1 and 2 has a strong feature for feature 1 and they have a similar feature vector. In the same manner, we inference the importance of words for each feature from the matrix $\boldsymbol{H}$. For the feature 1, the words 1, 2, 3, 4, 5 has weights 0.32, 0.32, 0.63, 0.63, 0 respectively. To sum up, the documents 1, 2 is similar and document 3 is not related to documents 1, 2 and the words 3, 4 are important terms of the feature 1 and terms 3, 5 are core words of the feature2.

This process remind the K-means clustering algorithm where K is a number of clustering. The k of NMF is the same as it. The degree of membership of documents for each clustering group is calculated by NMF. Besides, NMF can be analyse the importance of words for each clustering group.

3.4.3 Empirical analysis and results with Syllable Vector

Using HKIB-20000, we run Pearson correlation for cosine similarity which is applied NMF. We set a rank $k = 1000$. For document number 5222, we calculate the cosine similarity for all documents.

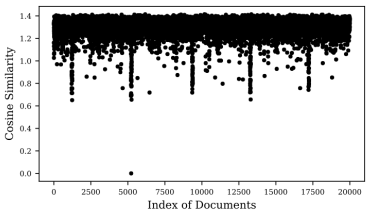


Figure 3-23: Syllable-1

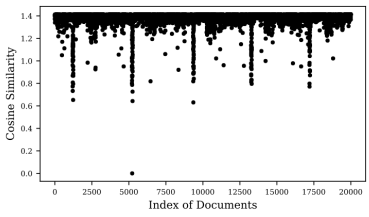


Figure 3-24: Syllable-2

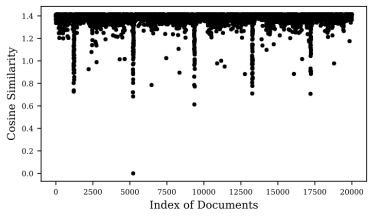


Figure 3-25: Syllable-3

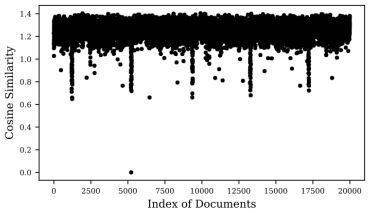


Figure 3-26: Syllable-1-All

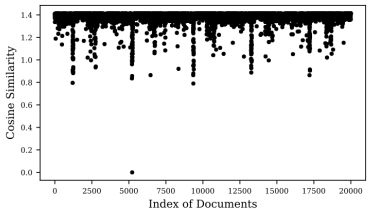


Figure 3-27: Syllable-2-All

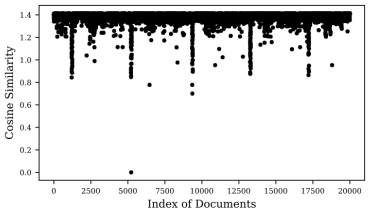


Figure 3-28: Word

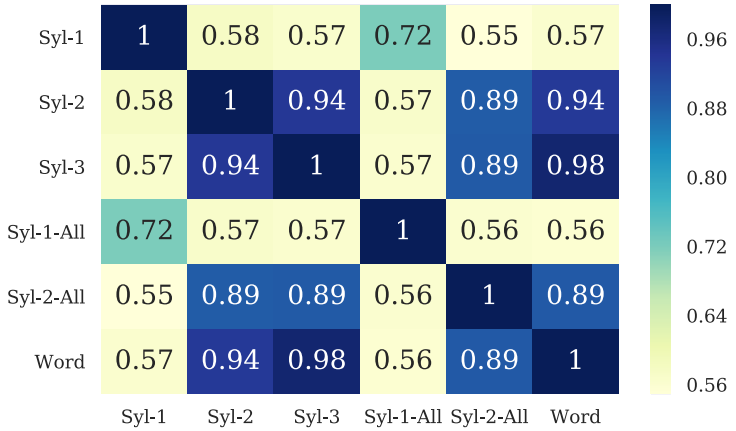


Figure 3-29: NMF* heat map by Pearson correlation for all documents

NMF conducts with the library *Scikit-learn*[4] and the library *nimfa*. For the convenience, NMF means the result of *Scikit-learn*, NMF* means the result of *nimfa*.

	Syllable-1	Syllable-2	Syllable-3	Syl-1-All	Syl-2-All	Word	Total
Dimension	1,266	33,639	59,574	1,379	57,541	61,176	-
Memory	194M	5.1G	8.9G	211M	8.6G	9.2G	-
NMF Time(s)	6353.91	17301.39	25781.61	6198.78	25535.54	26243.12	107414.35
NMF Time(h)	1.8	4.8	7.2	1.7	7.1	7.3	29.8
nimfa Time(s)	3005.49	50440.10	95940.48	3070.27	106440.50	140282.80	399179.64
nimfa Time(h)	0.83	14.01	26.65	0.85	29.57	38.97	110.88

Table 3-7: Consuming time for Cosine similarity between 20 sample set and all other documents

[4] NMF in the module *sklearn.decomposition*

3.5 Text Clustering

Clustering is about creating groups of data sets that are characteristic of given data. Especially text clustering is a important part of text mining. The reason is that the some features such as category, topic, tag can be grouped by clustering. Basically clustering is a unsupervised classification that designates the creation of classes or groups of a certain number of similar objects without prior knowledge [37]. Simply, this is to find proper groups of similar objects in the data. This area has been widely studied, and there are a tremendous number of methods. In this chapter, we focuse on how well the proposed method works with the conventional methods. Our suggestion is a core measurement about Korean text mining. It will be change totally the word-based process to the syllable-based in the conventional methods.

3.5.1 Evaluation of Text Clustering

Evaluation of clustering results is as difficult as the clustering itself. Clustering result has no answer unlikely text classification[5]. Then the evaluation of clutering is not clear and we need some idea for the validation.

Fundamentally, all clustering methods attept to maximize the following measures. [38]

Coherence. How similar are objects in the same cluster?
Separation. How far away are objects in different clusters?
Utility. How useful are the discovered clusters for an application?

Considering these core concepts, there are three kinds of criteria : *Internal criteria*, *External criteria*. [43]

External criteria means that we have some information about the labels.

To evaluate the performance of the proposed method, we use a *gold standard* which is generated by human with following evaluation tools.

3.5.2 Purity

Purity is a measure of the extent to which clusters contain a single class [28, 36]. First each cluster is assigned to the class which is most frequent in the cluster. And purity is measured by counting the number of correctly assigned documents and dividing by n.

$$\text{Purity} = \frac{1}{n}\sum_{q=1}^{k}\max_{1\le j\le l}(n_q^j) \tag{3.5.1}$$

where n is a total number of samples and n_q^j is a number of samples in the cluster q that belong to original class j $(1 \le j \le l)$.

[5] Classification is a supervised learning (labled data), Clustering is a unsupervised learning (unlabled data).

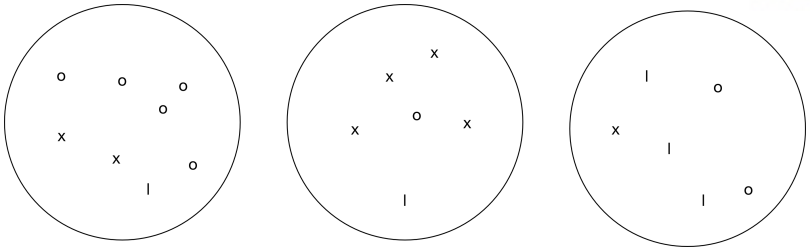


Figure 3-30: Example clustering set for purity

For instance, purity of the data set of Figure 3-30 is $\frac{1}{20}(5+4+3) = 0.6$. The larger the value, the better performance.

3.5.3 Precision and Recall

Measuring of the performance of supervised learning can start from build up the confusion matrix[6] which records the observed classified data and the predicted classified data.

		Predicted		Total
		Positive	Negative	
Observed	Positive	TP	FN	TP+FN
(Gold Standard)	Negative	FP	TN	FP+TN
Total		TP+FP	FN+TN	TP+FP+FN+TN

Table 3-8: A confusion matrix for binary classification

With this table, define the four kinds of the measurement(precision, recall, accuracy, f-measure). Consider the positive class of the predicted data in the first column of the confusion matrix. How many truly answer is in there? It is called the *precision*. In information retrieval contexts, the data can be divide the two types that the retrived documents (e.g., the list of documents produced by a web search engine for a query) and the relevant documents (e.g., the list of all documents for a specific topic). In other words, the retrieved documents that can be represented by symbolic is TP+FP, the relevant documents is TP+FN.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \tag{3.5.2}$$

Let's look at the sight of the positive answer with the first row of the confusion matrix. How

[6] TP = True Positive, FN = False Negative, FP = False Positive, TN = True Negative

many predicted positive data of the data of true positive? It is known as *recall*.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|} \tag{3.5.3}$$

Precision is to measure the quality of predicted data only based on what our predictor claims to be positive, no matter how good or bad negative predicted the data. On the other hand, recall is to measure the as positive predicted result of the truly positive answer. It is also called *sensitivity* or *true positive rate*. In the opposite sense, there is *specificity*(true negative rate), but we do not consider it here.

Thirdly define *accuracy* using every term of the confusion matrix. Accuracy show the probability of the true value of the predicted label.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3.5.4}$$

However, higher accuracy does not guarantee overall better performance of the method. Thus a combination of measures gives a balanced validation of the method. [44]

The one of the solutions is F-measure [42]. It combines precision and recall like harmonic mean of precision and recall.

$$F = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{3.5.5}$$

The harmonic mean is more intuitive than the arithmetic mean when computing a mean of ratios. For example, any system such as face recognition has 1.0 precision and 0.2 recall. The performance of this system is very poor because this system has many case for the false recognition. However the arithmetic mean is 0.6, whereas the harmonic mean is $\frac{1}{3} \approx 0.33$. The harmonic mean reflects the smaller number as compared to the larger number. Thus the harmonic mean is a more reasonable metric for the poor performance.

And it is also known as *F1-measure*. The general form of F-measure is that [45]

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}} \tag{3.5.6}$$

We can control the weighted factor like that F_2-measure attach weight to recall rather than to precision and $F_{0.5}$-measure is a reversed case.

3.5.4 Evaluation

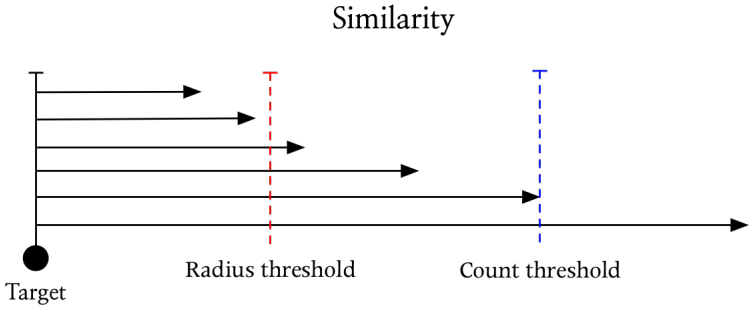


Figure 3-31: Example of threshold for constructing cluster

In order to evaluate the proposed method, we use a class-based Precision and Recall. We have the categories of documents by human as a gold standard (Fig 3-4). To make the group of documents, we use two rules. First rule is a count based threshold. For each document, we count the nearest neighborhoods. And second rule is a cluster radius based threshold. A visualization of the rule is Fig 3-31, 'target' is a target document and arrows means a similarity for each document from target document (0 near best). To grasp easily, refer Fig 3-32. We conduct the experiments based on the raw TF matrix.

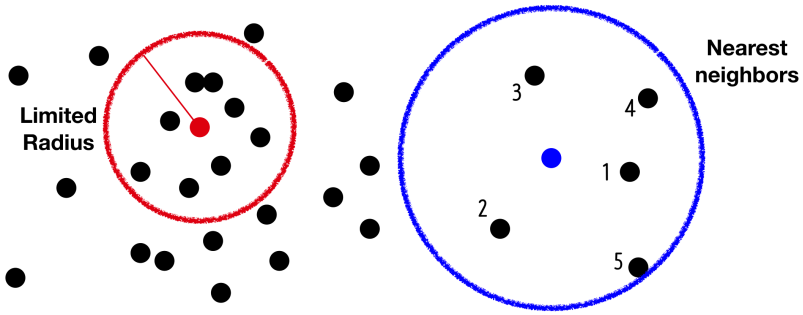


Figure 3-32: Example of threshold for constructing cluster

As we can see, the results are impressive. First, look at the case of count threshold (Fig 3-33). *Syl-1-All* and *Syl-1* missed some real answers. However, *Syl2*, *Syl-3* were almost as close to the case of *Word*. For more detail, you can see the right-side figure which is a expansion of left-side figure. *Word* has a best performance, followed by *Syl-3*, *Syl-2* and *Syl-2-All*.

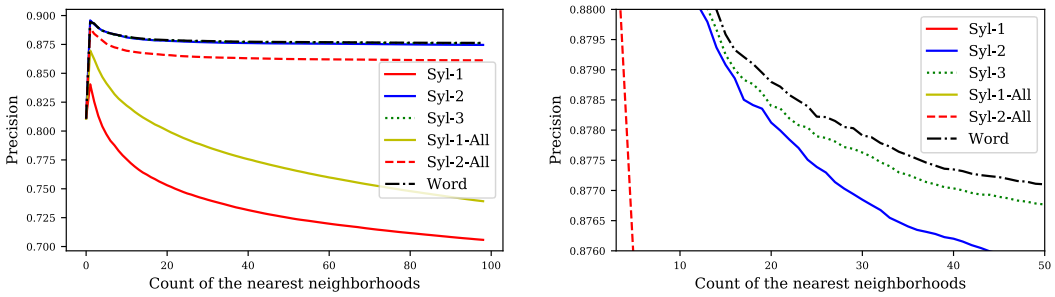


Figure 3-33: Precision for the count of the nearest neighborhoods

And the next thing is a case of radius threshold (Fig 3-34). Likewise, *Syl-1-All* and *Syl-1* has some errors for real answers. And *Word*, *Syl-3* and *Syl-2* are best, it's hard to distinguish superiority, followed by *Syl-2-All*.

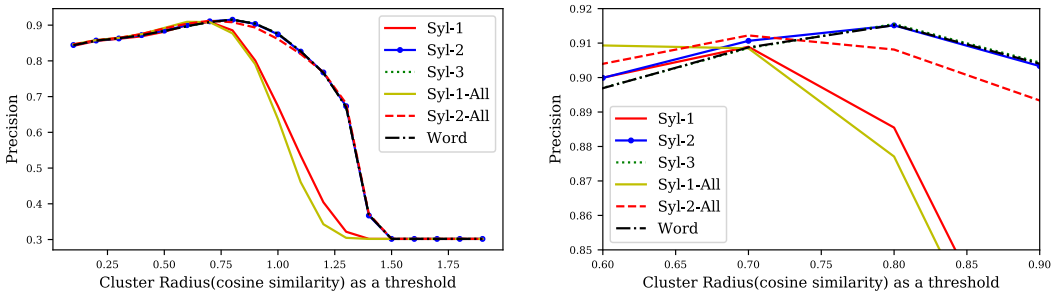


Figure 3-34: Precision for the cluster radius at each documents

In these experiments, we know the *Syl-2* has impressive performance compared to the *Word* and *Syl-2-All*. The next question is how performance will vary depending on how we apply the matrix operation. We will try it in the next section.

3.5.5 Top Ranking Matching

To compare the performance of each method(Basic, SVD, NMF*[7]), we select the count threshold as a evaluation case with $n = 5$. This is called an "Top Ranking Matching" and is depicted in the right-side of Fig 3-32 (Blue Circle). We deal with the two types of result that Precision and Speed.

Fig 3-35 show that the performance of *Syl-2, 3* and *Syl-2-All* is similar to *Word*. The interesting point is that *Syl-1* and *Syl-1-All* which has information loss has also good precision after the non-negative matrix factorization. Their speed is very fast compared to others (Fig 3-36). For more easy understanding, let's look at a combined graph of these factors. Fig 3-37 illustrates the performance of precision versus speed. *Syl-2* with LSI is the best. It would have resulted from LSI features which captures the synonym or polysemy. In other words, LSI is the

[7] NMF* is a result of the library *nimfa*. Refer Table 3-7.

best of the three. And *Syl-1-All* is also good choice if you want to save the time. *Syl-1* use also small time, but it has more information loss. It cause the lower precision than *Syl-1-All*. Interestingly, however, NMF seems to be filling that loss.

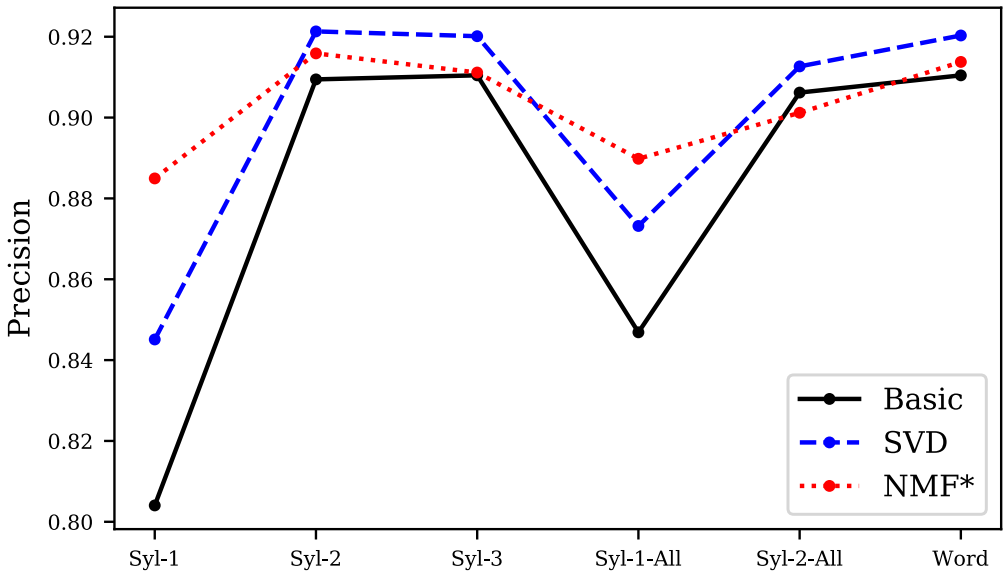


Figure 3-35: Precision of the top ranking matching ($n = 5$)

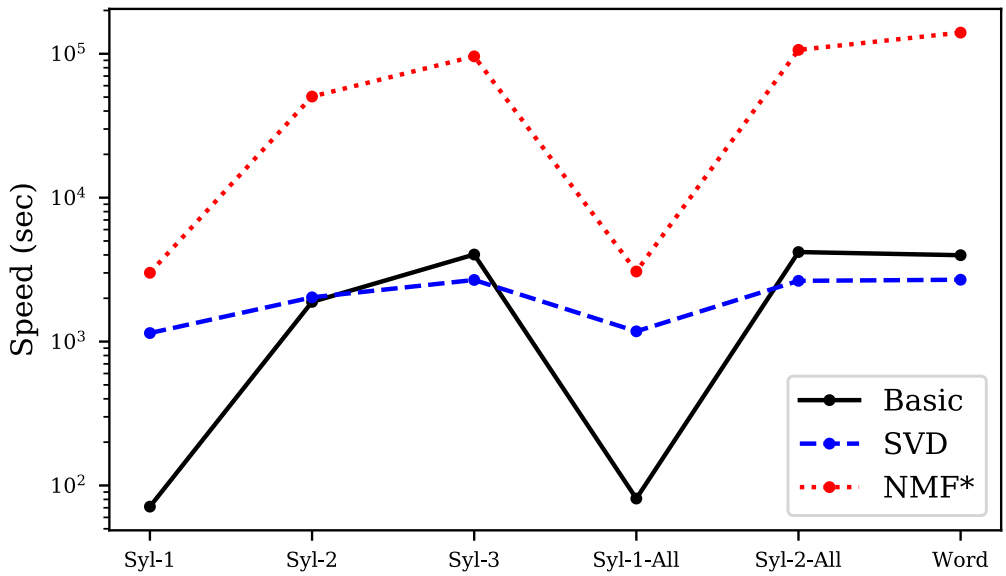


Figure 3-36: Speed of the top ranking matching ($n = 5$)

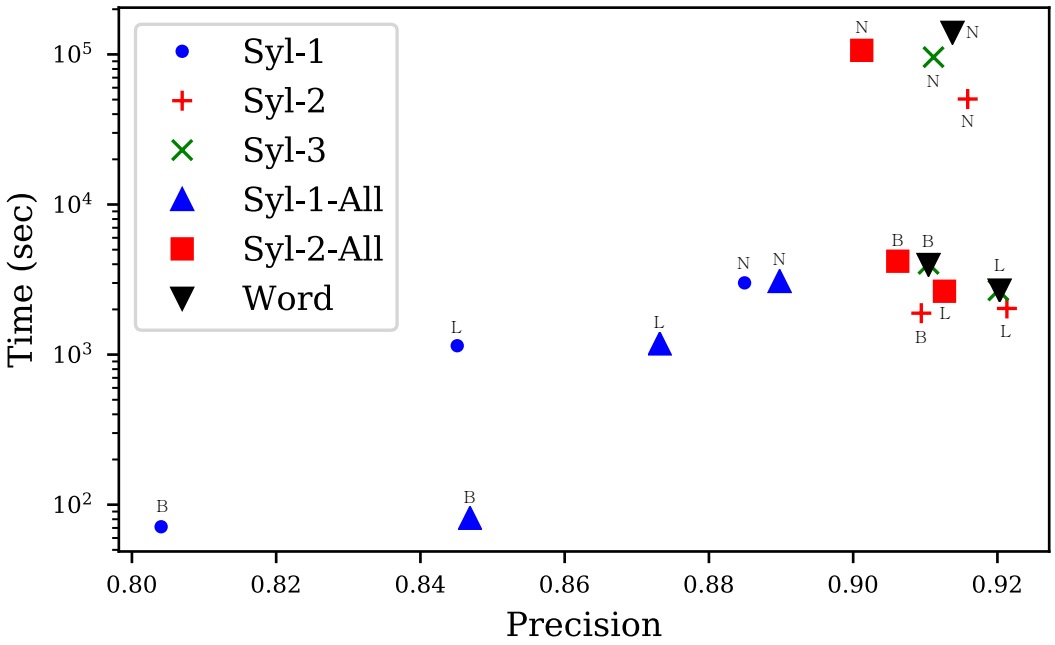


Figure 3-37: Precision vs Speed (B : Basic, L : LSI, N : NMF)

3.6 Results and discussion

Syl-2, *Syl-3* has very close performance with *Word*. This means that taking the n syllables only has a sufficiently large effect. And the advantage is a speed. *Syl-2* is about 1.5 times faster on average than *Word*. The reason is the dimension size of matrix, they are almost half different.

Syl-1-All is good if we want to speed up even if we lose some precision. This extremely reduces the size of the matrix compared to the word case. *Syl-1* has large information loss, then has low precision. However, *Syl-1-All* overcomes some of these drawbacks and shows good performance.

Numerous algorithms have surfaced that take the program to the next level. We can apply the matrix operation of new concept such as adaptive tf-idf, genetic algorithm, etc. For example, our experiments conducted with the raw TF matrix. When we apply the TF-IDF, correlation has been improved (Fig 3-38 $\sim$ 3-43).

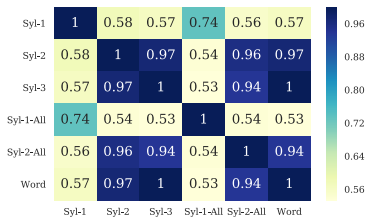


Figure 3-38: Corr Basic

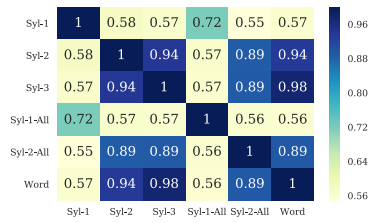


Figure 3-39: Corr NMF*

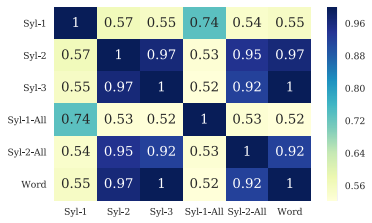


Figure 3-40: Corr SVD

We expect that the application of these new methods will make our proposed method (*Syllable-n* vector, *Syllable-n-All* vector) more effective.

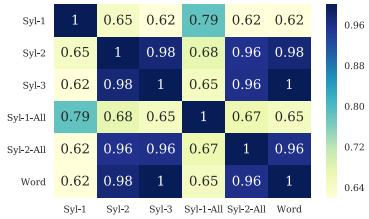


Figure 3-41: Corr Basic with tf-idf

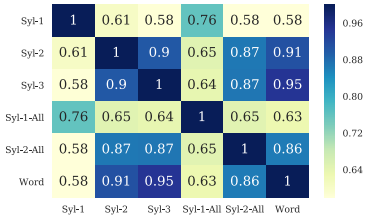


Figure 3-42: Corr NMF* with tf-idf

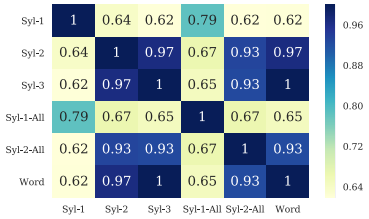


Figure 3-43: Corr SVD with tf-idf

And this is not the only method for Korean documents. Clustering of English documents translated into Korean may improve clustering performance of English documents due to the characteristics of Korean.

4

Joint Analysis of Text and Relational Data

We have done the text analysis with only the informaion of the document itself. In chapter 2 and chapter 3, all concept use the document itself as features. That is, text information has an independent relationship between documents. However documents has more information between them. We call it the *meta-data*. Metadata means "data about data". This is too abstract, but it is true. In other words, metadata is defined as the data providing information about one or more aspects of the data [46]. Metadata is divided three types that descriptive, structural and administrative metadata. In this chapter, we focus the structural metadata, especially relational data. We work the joint analysis of text and relational data of text.

4.1 Word2Vec

Vector Space Models (VSMs) have been used for a long time as a means to express textual information mathematically. Latent Semantic Indexing (LSI) and Latent Dirichlet Allocation (LDA) have been the traditional method but recently Word2Vec using Artificial Neural Networks(ANN) has been regarded highly as a new method.

Word2Vec is part of a word embedding method in which represents words in the form of vectors and is an unsupervised learning algorithm that is based on neural networks that automatically learn the relationship between words. Word2Vec takes into consideration the word's meaning and context when expressing it as a vector and this is done so under the distributional hypothesis in linguistics. Distributional hypothesis means that words with similar distribution will have similar meaning. Having similar distribution means that the words will tend to appear in the same context. For example, in "Burger King's hamburger tastes good" and "McDonald's hamburger tastes good" Burger King and McDonald's are surrounded by the same context and thus will share the same meaning. Word2Vec was created by a team of researchers

led by Mikolov in 2013 [47] and was applied into various algorithms such as Sentence2Vec, Paragraph2Vec, Doc2Vec, LDA2Vec etc.

The basic idea is that similar meaning words can be found in similar positions in context. This can be calculated in two ways: Continuous Bag of Words (CBOW) or a Skim-gram. With CBOW, a contextual set of C number of terms are entered into the input layer and a set of target terms are entered into the output layer and is trained with those set answers (Figure 4-1). On the other hand, with a Skip-gram we enter the target terms and train the neural networks to then predict the contextual structure that surrounds a certain term (Figure 4-2). C is a window parameter for how far from the target term contextual terms will be taken into account.

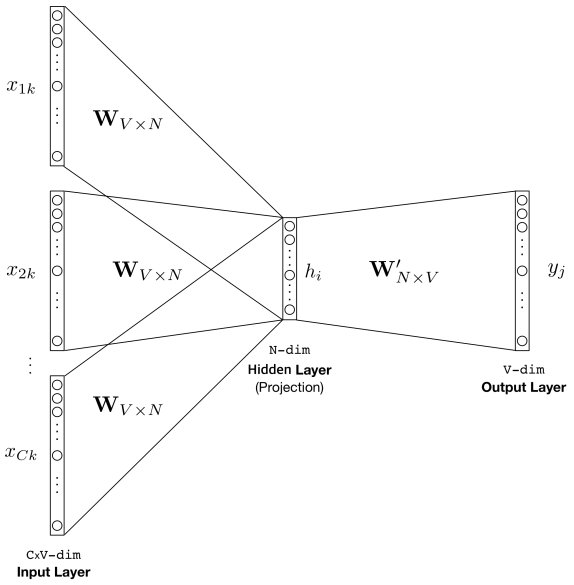


Figure 4-1: CBOW model

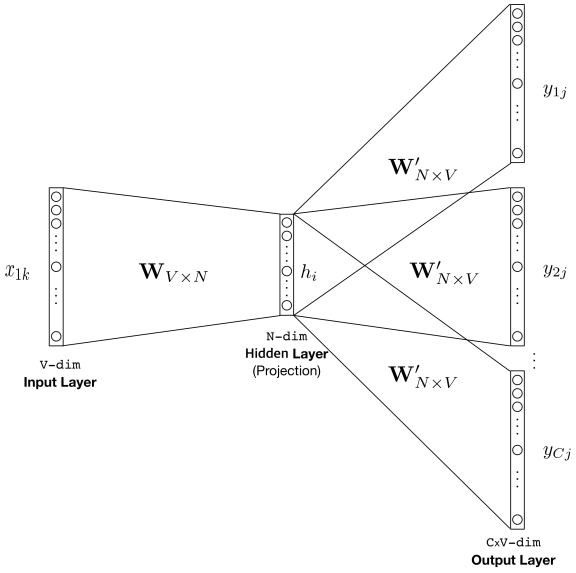


Figure 4-2: Skip-gram model

These architectures work well with legal text data. The results of learning about 76,000 Korean case laws using Word2Vec can be found in Table 4-2. For example, we will learn the following data. First we analyze the morpheme of the selected data and break it down into units of terms with meaning. Here, we allocate the units as nouns.

Num	Sentence / Terms
1	When a high school teacher strikes a student on the left cheek as punishment for wrongdoing and causes the student to die from falling after the strike → High School, teacher, student, cheek, punishment, wrongdoing, strike
2	To strike the cheek of a drunk troublemaker twice is not a violation of social norms → Cheek, troublemaker, violation, social norms

Table 4-1: Example for learning data set

Let us define the number of context terms C as 1. The term list for each sentence will be then: [High School, teacher, student], [teacher, student, cheek], [student, cheek, punishment], [cheek, punishment, wrongdoing], [punishment, wrongdoing, strike], [cheek, troublemaker, violation] ,[troublemaker, violation, social norms].

The term in the middle of each team of words will become the target term and the terms beside the target term will become the contextual terms. This is then used as a learning model under the architecture of CBOW or Skip-gram.

Target Word	**법**	**판결**	**재판**	**빰**	**대학**
Similar Words	법률 0.583	무죄 0.377	소송 0.47	머리채 0.345	대학교 0.51
	특별법 0.469	항소 0.366	재심 0.428	주먹 0.342	학교 0.485
	헌법 0.426	심판 0.364	구속 0.422	구타 0.32	교육 0.379
	사법 0.42	심결 0.333	항소심 0.385	멱살 0.306	대학인 0.37
	관리법 0.415	재판 0.324	상소 0.371	정강이 0.293	학원 0.343
	약사법 0.409	형 0.318	심판 0.346	행패 0.29	대학원 0.337
	의료법 0.408	결정 0.315	제기 0.339	반말 0.288	학과 0.331
	수산업법 0.408	원판결 0.304	판결 0.324	폭행 0.286	학생 0.328
	거래법 0.393	환송 0.283	재판장 0.315	원산폭격 0.284	교수 0.325
	부칙 0.39	판단 0.279	법원 0.313	뒤통수 0.284	교원 0.318

Table 4-2: Example for legal korean case sentence data

Word2Vec follows the features :

- a learning set for only words
- use only context(just beside) words

Word-based methods have limitations due to the problems of the word itself. Thus we need to use other information. To apply the meta data between documents, we take advantage of Word2Vec to design a new architecture which has bipartite data set. Let's take a look at the next chapter.

4.2 Heterogeneous Word2Vec

Heterogeneous means different types data set for input, output layers. They have similar vector position if they has similar relationships. To understand the architecture of Heterogeneous Word2Vec, let's take for example. We just consider the skip-gram architecture in here. Basically Word2Vec start from the one-hot encoding.

Suppose that we have a following number sequence:

$$\text{data} = \{0, 1, 2, 3, 4, 5, 6\} \tag{4.2.1}$$

The one-hot encoding changes the word to the zeros vector except for one element. For the word '0', it has a one-hot encoding vector $[1, 0, 0, 0, 0, 0, 0]$ which has a vocabulary size of data set as a dimension. These vectors are unique and sparse.

Word	One-hot encoding vector
1	[1, 0, 0, 0, 0]
2	[0, 1, 0, 0, 0]
3	[0, 0, 1, 0, 0]
4	[0, 0, 0, 1, 0]
5	[0, 0, 0, 0, 1]

Table 4-3: One-hot encoding vectors for input layer

Word	One-hot encoding vector
0	[1, 0, 0, 0, 0, 0, 0]
1	[0, 1, 0, 0, 0, 0, 0]
2	[0, 0, 1, 0, 0, 0, 0]
3	[0, 0, 0, 1, 0, 0, 0]
4	[0, 0, 0, 0, 1, 0, 0]
5	[0, 0, 0, 0, 0, 1, 0]
6	[0, 0, 0, 0, 0, 0, 1]

Table 4-4: One-hot encoding vectors for output layer

	Target	Context	Learning Data Set
Word	1	0, 2	[1, 0] , [1, 2]
	2	1, 3	[2, 1] , [2, 3]
	3	2, 4	[3, 2] , [3, 4]
	4	3, 5	[4, 3] , [4, 5]
	5	4, 6	[5, 4] , [5, 6]

Table 4-5: Converted learning data set

In the architecture of Word2Vec, we convert the given data to a set of learning data like Table 4-5. The target word is a object word for learning and the context word is a supervised data for the object word. We follow the general neural networks(NN) for a hidden layer and set the six nodes for hidden layer in this example.

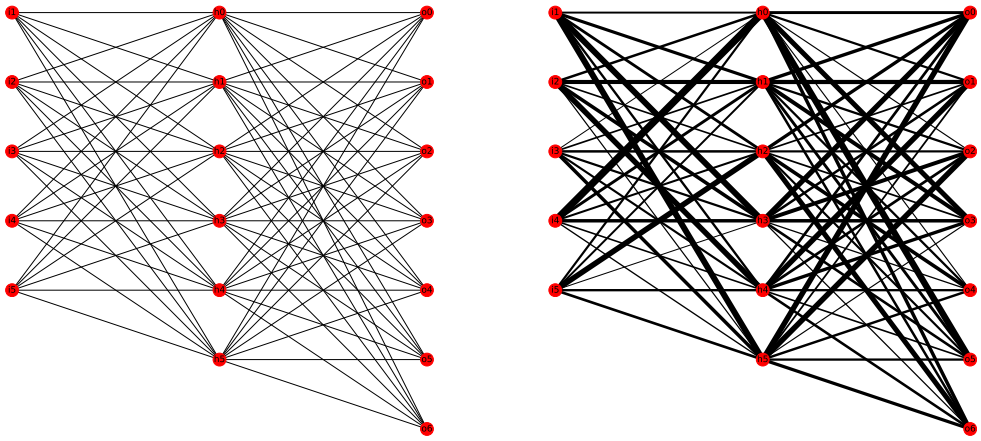


Figure 4-3: Example of Skip-gram model (left: initial NN structure, right: learned weighted NN structure

In the process of learning, the form of the data is [input, output] like that $[1, 0]$, $[1, 2]$, $[2, 1]$, $[2, 3]$, $[3, 2]$, $[3, 4]$, $[4, 3]$, $[4, 5]$, $[5, 4]$, $[5, 6]$ from Table 4-5. At the beginning, the matrices $\boldsymbol{W}_{V\times N}$, $\boldsymbol{W}'_{N\times V}$ are randomly initialized by the normal distribution. For convenience, we write the weighted matrices as $\boldsymbol{W}_1$, $\boldsymbol{W}_2$ respectively. The distribution of initial value of matrix follows the normal distribution. However after learning the distribution of values is biased like Figure 4-15, 4-16.

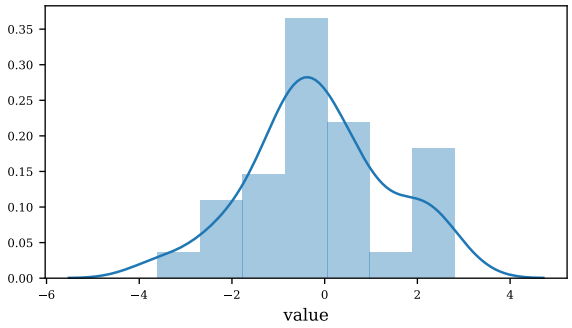


Figure 4-4: Distribution for W_1 matrix

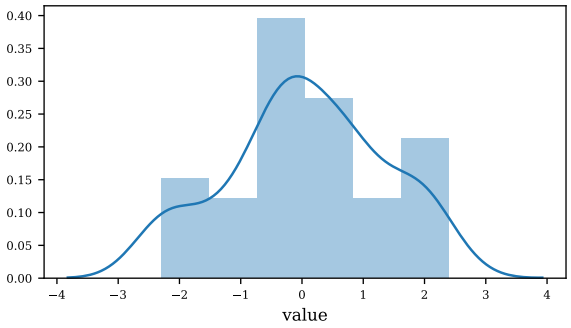


Figure 4-5: Distribution for W_2 matrix

After learning, the well learned neural networks structure can be represented like Figure 4-6. Each row of $\boldsymbol{W}_1$ is the N dimensional vector representation $\boldsymbol{v}_w$ of the input word $\boldsymbol{x}_w$. Because the dot product of a one-hot encoding vector and the weighted matrix means a embedded word vector of the word such as lookup table.

$$\boldsymbol{v}_w = \boldsymbol{W}_1^T \boldsymbol{x}_w \tag{4.2.2}$$

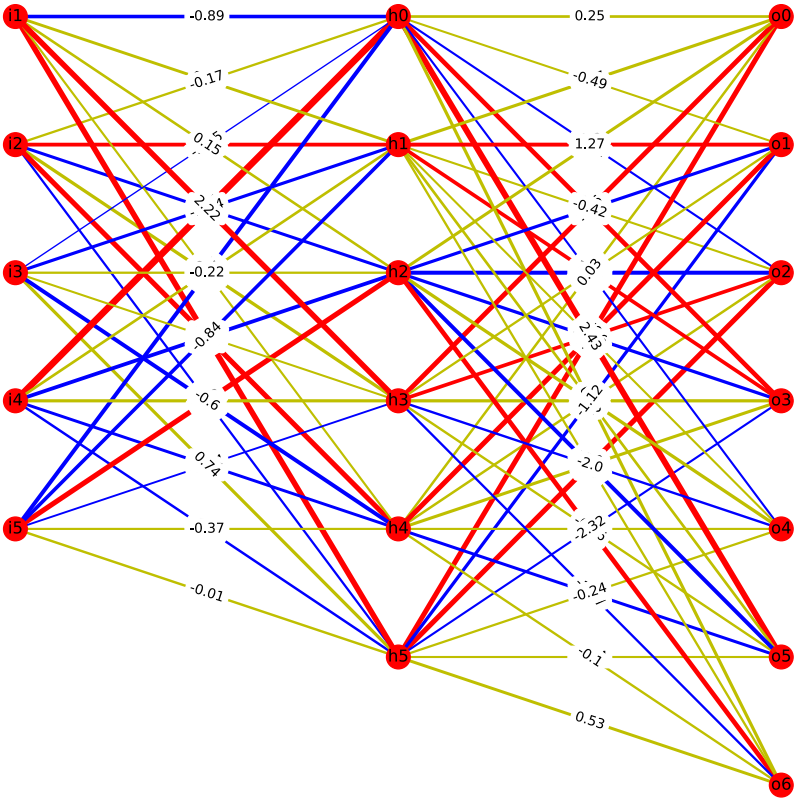


Figure 4-6: Example for the neural networks structure after well learned (Edge color red: value > 1, blue: value < -0.5, yellow: others)

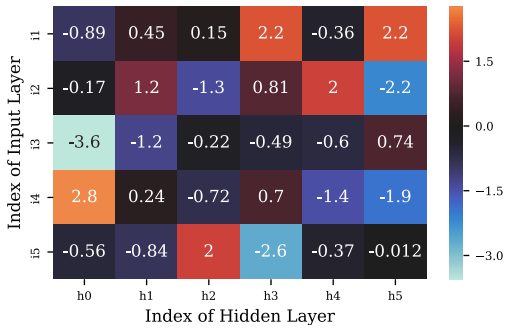


Figure 4-7: Heatmap for W_1 Matrix

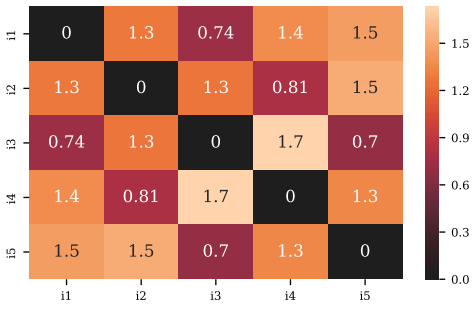


Figure 4-8: Similarity of Input Nodes

To show the tendency of well learning for large data, we set the number sequence $0 \sim 108$ with $C = 4$, $N = 50$. Fig 4-12 and Fig 4-14 show the strong relationships at the diagonal elements.

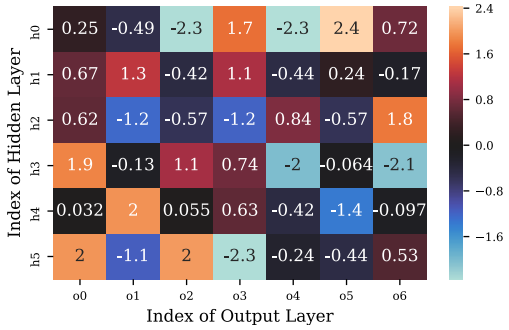


Figure 4-9: Heatmap for W_2 Matrix

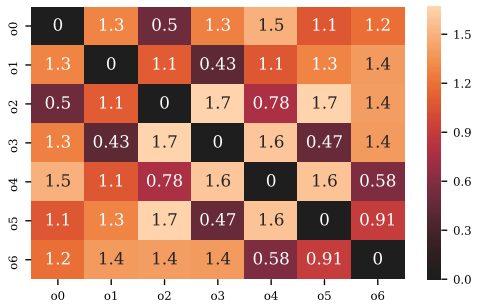


Figure 4-10: Similarity of Output Nodes

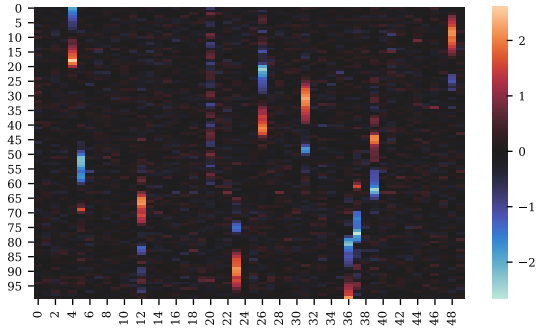


Figure 4-11: Heatmap for W_1 Matrix for a large set

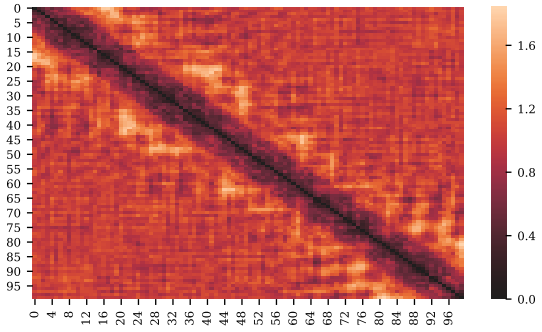


Figure 4-12: Similarity of Input Nodes for a large set

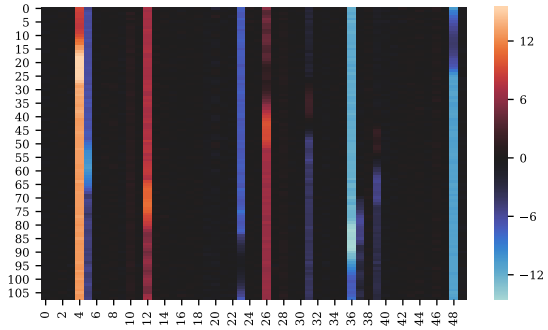


Figure 4-13: Heatmap for W_2 Matrix for a large set

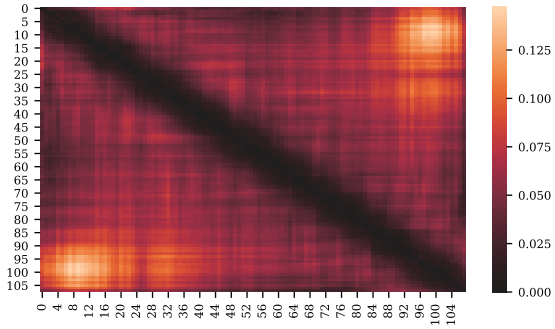


Figure 4-14: Similarity of Output Nodes for a large set

4.3 Law2Vec

Legal information comprises mainly of legislation and case law. Case law represents court decisions upon cases that question the interpretation of certain aspects of law. Every case law becomes a precedent for future cases and act as the set answer to legal questions. Furthermore, there is no case law that does not depend on law itself. The court will always be addressing an aspect of law, thus case law will always consist of references to legislations and most likely will refer to one or more precedent cases as well. The goal is to use such references between case law

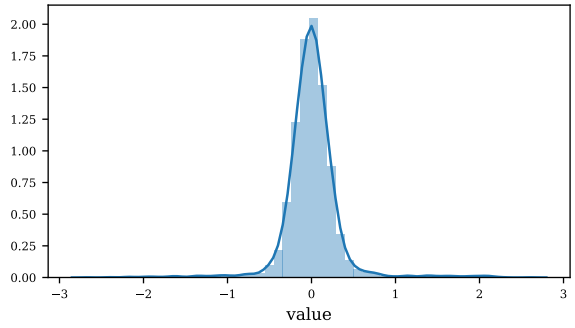


Figure 4-15: Distribution for W_1 matrix for a large set

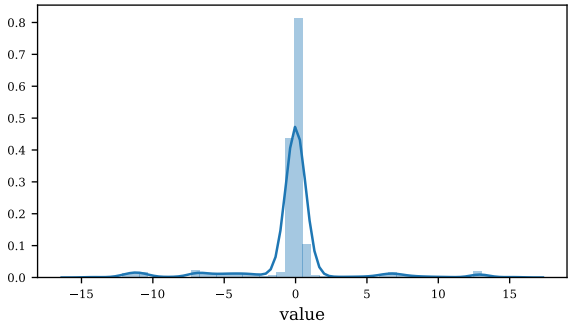


Figure 4-16: Distribution for W_2 matrix for a large set

to extract meaning that can then be used to predict relationships between legal data.

Ways to calculate the similarity between documents have been researched for a long time. In particular, research for calculating the similarity of documents in the field of legal data is a necessary tool in order to develop efficient legal data processing tools such as legal search engines. To locate the case law or legislation related to one's legal work or case is a necessary task in the area of legal research and case analytics. Up to now, this task has been prosecuted using a keyword-matching method that calculates a similarity score of each document to the keyword and lists the results of the search in the order of that score. However, this keyword matching method only operates for that exact keyword and dismisses synonyms or other related keywords to that keyword and therefore is not able to calculate similarities of documents properly. To solve this problem, many researches have been conducted such as the Latent Semantic Indexing method that calculates approximated matrix to find synonyms. These researches, however, are still focused on keywords and have difficulty taking into account the semantics of each legal document, especially of case law.

Case law consists of many different material facts and reasoning. For example, 2007후3806 is a case in which the scope of patent rights is discussed and deals with the patent title "다수의 자외선 램프를 구비하는 수처리장치"(WATER-CLEANING APPRATUS HAVING A PLURALITY OF UV LAMPS). However, the key words "UV lamps"', "water-cleaning apparatus", "plurality" do no good for calculating the similarity of the semantics of the case.

We define the semantics of a case by the issue and the arguments of the case, and when calculating the similarities of such legal topics, there are many unneeded keywords that appear. To deal with these distractions methods such as Latent Dirichlet Allocation (LDA) and Word2Vec is applied, but they are not enough to solve the core problem.

We propose a new architecture that is able to show the similarities between the fundamental elements of a case which are case law and legislation. We call this Law2Vec and is composed of three different forms using case law and legislation.

4.3.1 Case - Legislations(CL)

Law does not rewrite the same information twice. It simply reuses an already ruled interpretation of a legislation and applies it to the current decision. A judge will look to the applicable legislation of his case and reason according to such legislation. Therefore, a reference to legislation will always exist in case law and we wish to learn this relationship between case law and legislation through the architecture of Word2Vec.

Legislations are divided into articles and sections that all represent a specific legal topic and case law allocates and applies such legal topic to cases. Therefore, the process of searching for case laws with similar context can be learnt through the application relationships between legislations.

In this section, we wish to show the possibility of a similarity search that takes into account the semantics of case law through the contextual learning of the relationship between case law and legislations.

There are two kinds of data that cases and cited legislations by case (Fig 4-17). And the data has also two types meta data (Table 4-6).

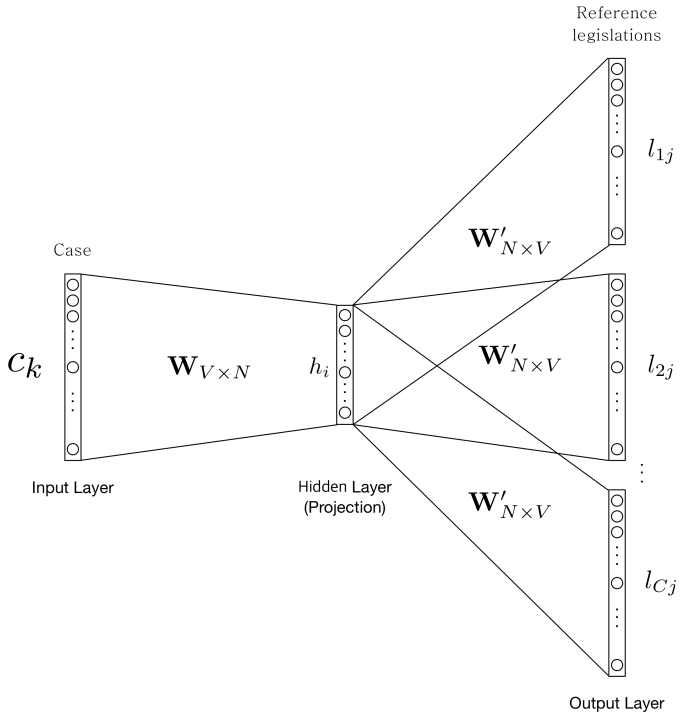


Figure 4-17: CL model

4.3.2 Case - Cases(CC)

Cases represent the court's judgment for a specific legal issue. If a court has already ruled how a legislation should be interpreted or how a specific legal issue should be judged, the court will cite a precedent case as it applies it to the current case. Precedent means that a principle or rule

established in a previous legal case has, depending on the legal system, binding or persuasive powers over a lower court when deciding a case with similar issues or facts, therefore a case usually always cites a previous precedent case. We propose a Case-Case Model that takes into account such citation relationships between cases (Fig 4-18).

For example, case 99두9902 cites the cases shown in Figure 22. These case citation relationships are learnt exactly as the CL Model was learnt.

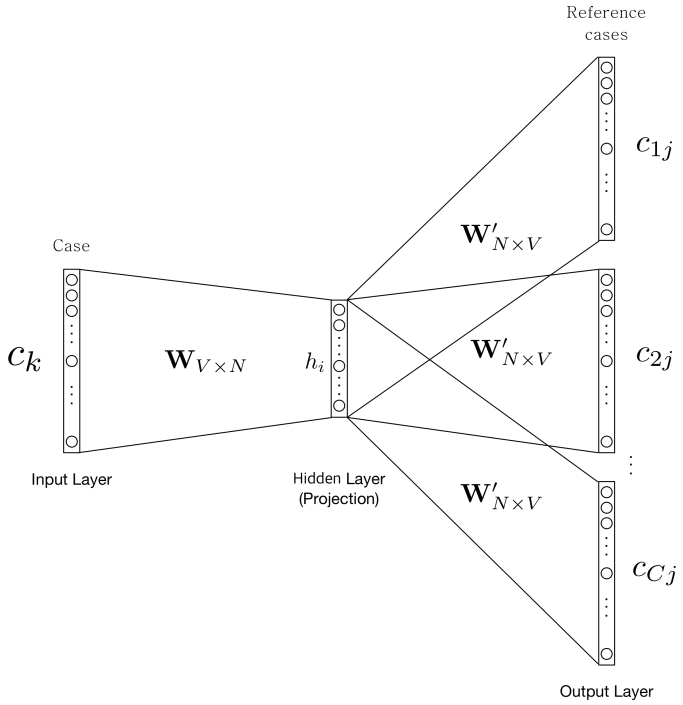


Figure 4-18: CC model

4.3.3 Case - Legislations, Cases(CLC)

Legislations are the core foundation of each legal case whilst precedent cases help to guide each case as to what direction each specific legislation must be interpreted. Therefore, cases will always cite legislations and precedent cases and this combination of two sets of data can be learnt through our CLC Model (Fig 4-19)

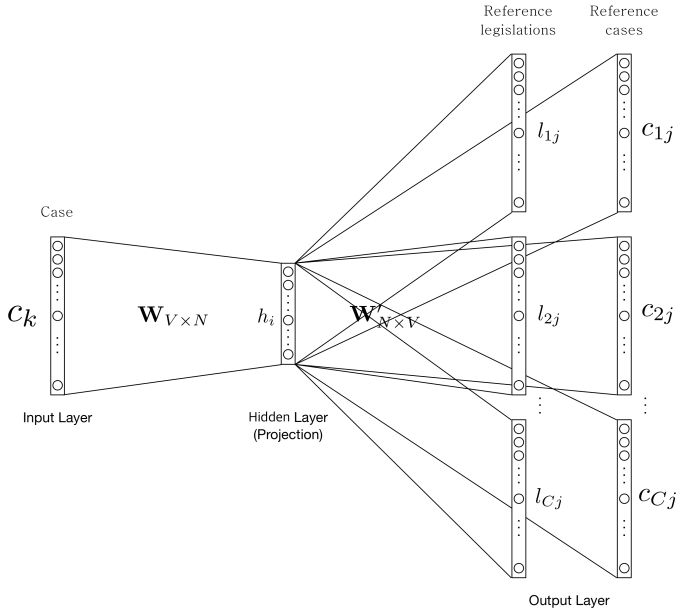


Figure 4-19: CLC model

4.3.4 Results

To compare the quality of different versions of Law2Vec, we test a small sample data set and a large sample data set.

4.3.4.1 Task description

To test the quality of the above proposed methods we select 5 sample cases and learn their citation relationships. In the Fig 4-20, labeled nodes are target cases and other nodes are cited objects which blue color means the case, red color means legislation and red lines are example of cited relationships for one case.

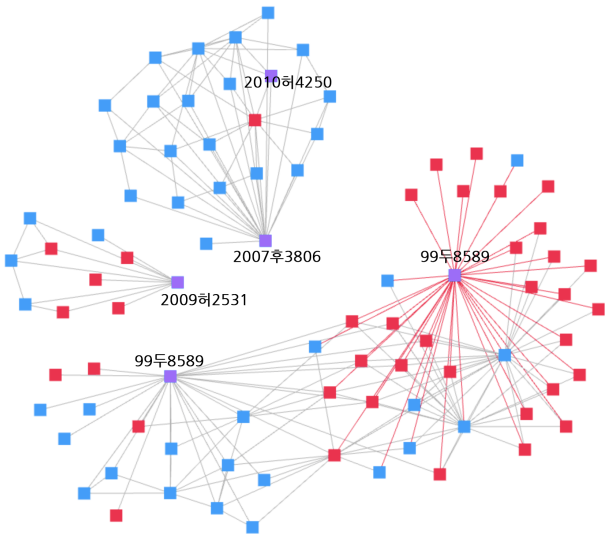


Figure 4-20: Citation relations of the given cases (blue node: case, red node: legislation)

We obtain a sample set of data that have been checked by a lawyer to be similar and related in material facts and legal issues. The answer sets are as follows: 1. Case 99두9902 and 99두8589 2. Case 2007후3806 and 2010허4250 3. Case 2009허5231

We apply this answer set to the three models CL, CC and CLC and compare the results.

Case Number	Cited legislations
99두9902	환경 · 교통 · 재해등에관한영향평가법_4, 환경영향평가법(구)_18, 환경영향평가법(구)_19, 환경영향평가법(구)_16, 환경영향평가법(구)_17, 환경 · 교통 · 재해등에관한영향평가법_21, 환경 · 교통 · 재해등에관한영향평가법_20, 행정소송법_27, 환경영향평가법(구)_4, 환경영향평가법(구)_1, 환경 · 교통 · 재해등에관한영향평가법_17, 환경 · 교통 · 재해등에관한영향평가법_19, 환경 · 교통 · 재해등에관한영향평가법_1
99두8589	먹는물관리법(구)_2, 관광진흥법(구)_2, 관광진흥법_53, 환경영향평가법(구)_8, 관광진흥법시행령(구)_22, 관광진흥법(구)_1, 환경영향평가법(구)_19, 환경영향평가법(구)_16, 행정소송법_1, 환경 · 교통 · 재해등에관한영향평가법_21, 환경정책기본법(구)_10, 행정소송법_27, 관광진흥법_2, 환경영향평가법(구)_9, 환경영향평가법(구)_4, 환경영향평가법(구)_1, 환경정책기본법(구)_5, 환경정책기본법(구)_1, 환경정책기본법(구)_7, 환경정책기본법(구)_6, 환경 · 교통 · 재해등에관한영향평가법시행령_2, 환경 · 교통 · 재해등에관한영향평가법_17, 관광진흥법(구)_25, 환경 · 교통 · 재해등에관한영향평가법_1, 환경 · 교통 · 재해등에관한영향평가법_6, 환경 · 교통 · 재해등에관한영향평가법_4, 환경 · 교통 · 재해등에관한영향평가법_5, 환경영향평가법시행령(구)_2, 먹는물관리법(구)_5
2007후3806	특허법_135
2010허4250	특허법_97, 특허법_135
2009허2531	상표법_3, 상표법_73, 상표법_71, 상표법_8, 상표법_7

Table 4-6: Example for sample target cases with cited legislations

4.3.4.2 CL

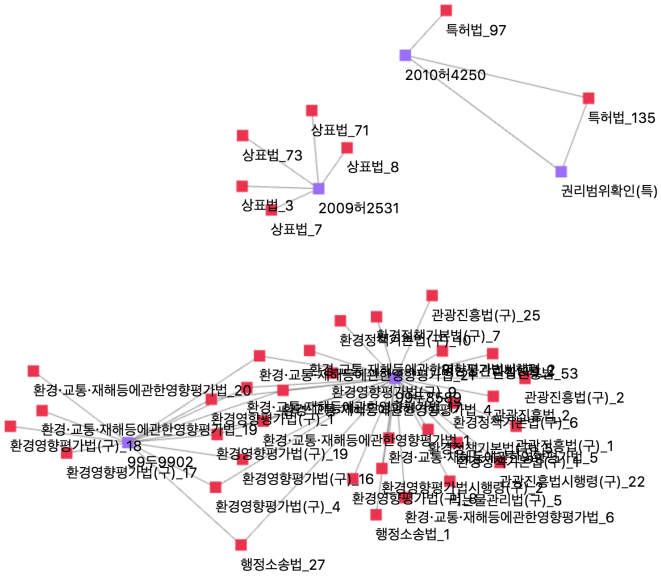


Figure 4-21: CL Citation network of Samples

The results of learning only the cited legislations of the selected 5 sample cases is shown is Fig 4-21. As seen in Fig 4-23, the similarity of each sample case was well learnt. We can also see that the legislations in matrix W_2 was well learnt according to each of its legal characteristics (Fig 4-25).

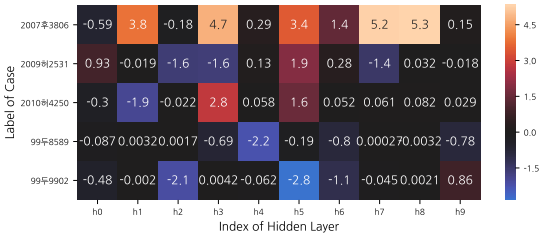


Figure 4-22: Heatmap for W_1 Matrix for CL

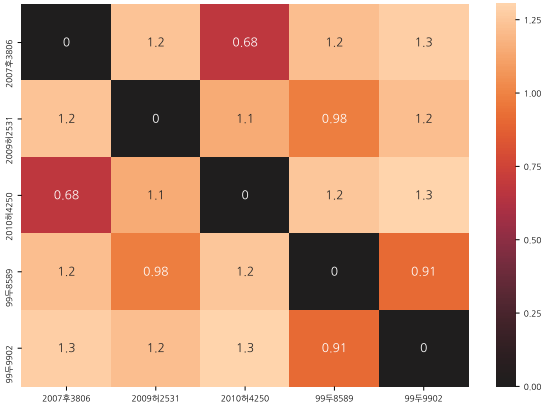


Figure 4-23: Similarity of Input Nodes for CL

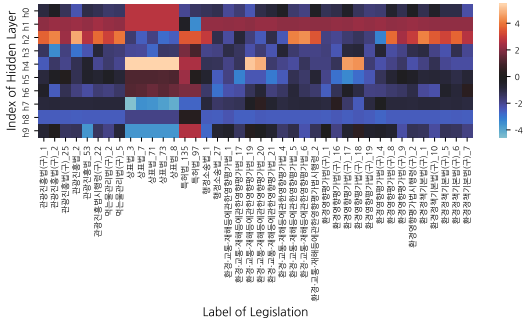


Figure 4-24: Heatmap for W_2 Matrix for CL

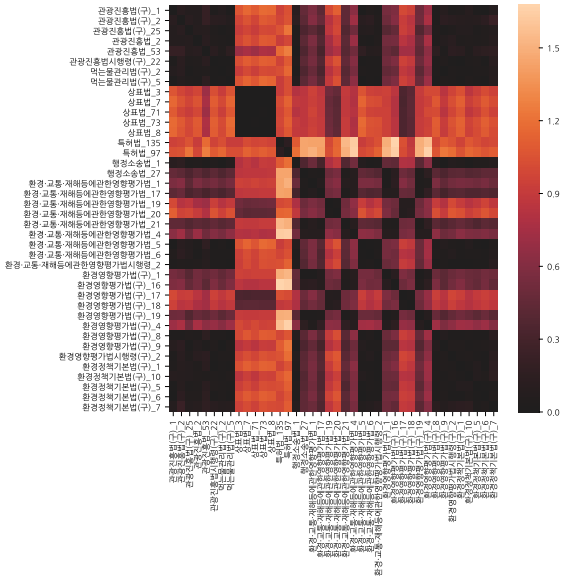


Figure 4-25: Similarity of Output Nodes for CL

4.3.4.3 CC

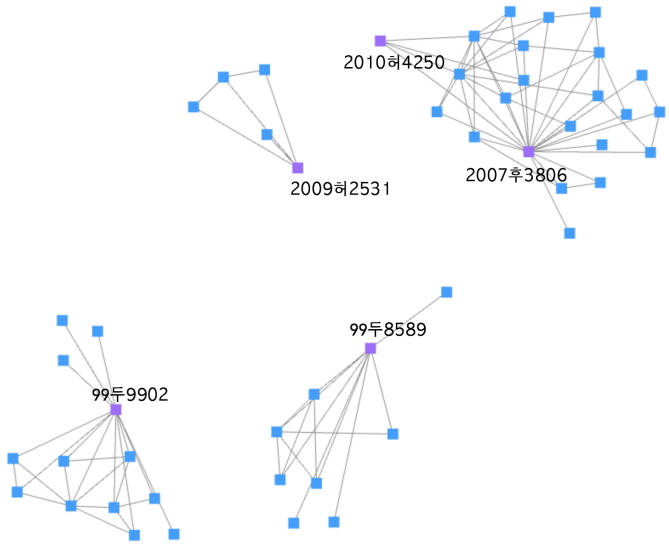


Figure 4-26: CC Citation network of Samples

Data cited between case to case can be presented as in Fig 4-26. It is worth noting that the answer set cases 99두9902 and 99두8589 are not linked with a line as they had been in the previous CL model. As a result, apart from case 2010허4250 and 2007후3806 showing a high similarity value (Fig 4-28), the other cases show a low, non distinct, similarity value.

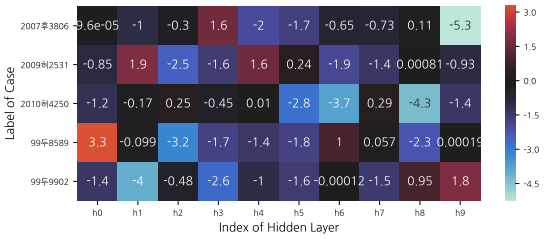


Figure 4-27: Heatmap for W_1 Matrix for CC

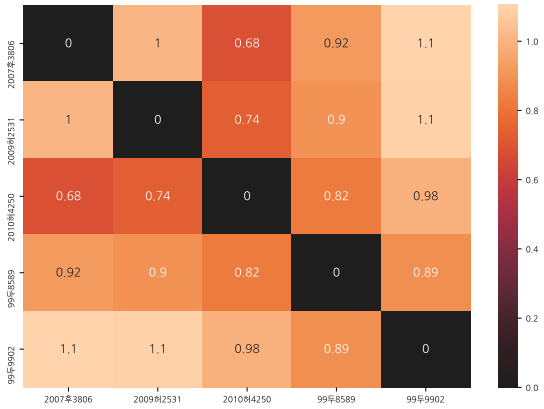


Figure 4-28: Similarity of Input Nodes for CC

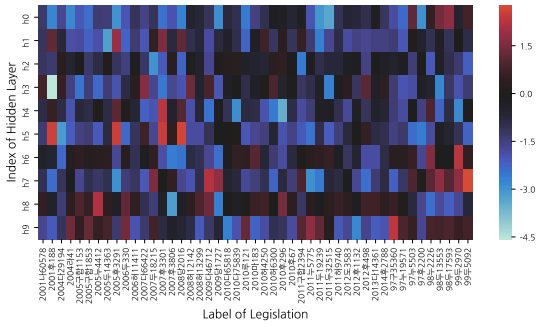


Figure 4-29: Heatmap for W_2 Matrix for CC

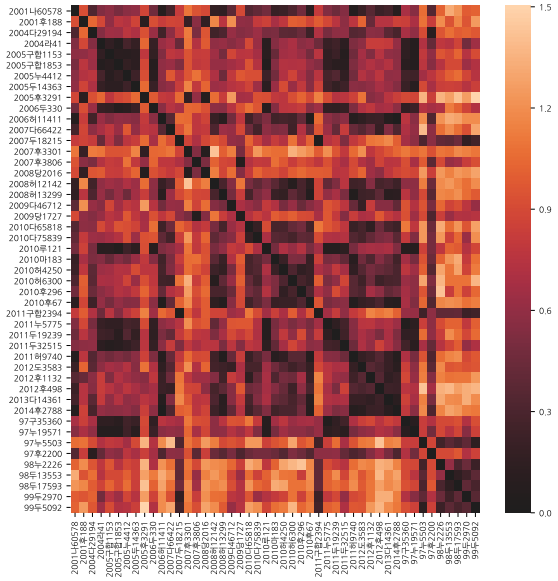


Figure 4-30: Similarity of Output Nodes for CC

4.3.4.4 CLC

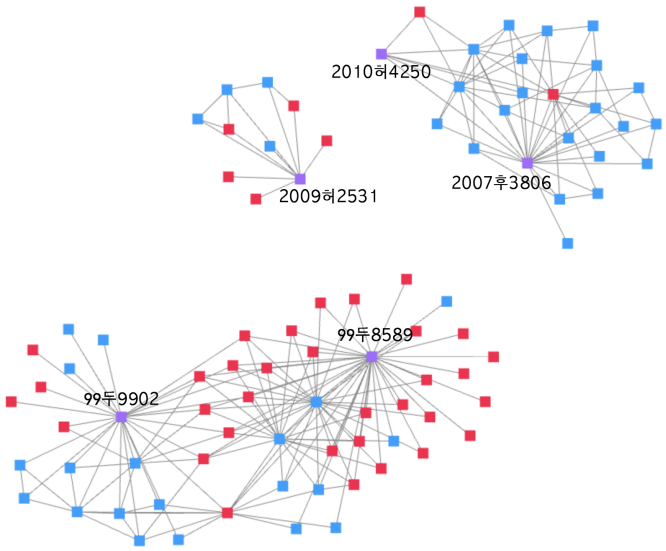


Figure 4-31: CLC Citation network of Samples

The last method learns the citation relationship of both cases and legislations of a case. The results show that the learning went very well and even the aspects that were not well learnt in the CC Model were successfully recognized in this model.

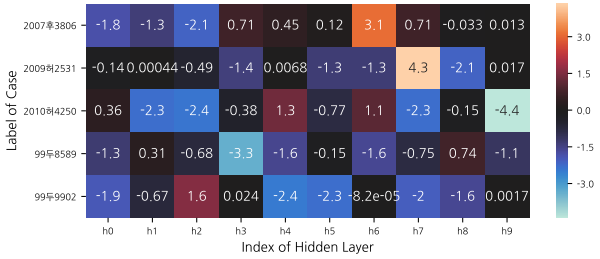


Figure 4-32: Heatmap for W_1 Matrix for CLC

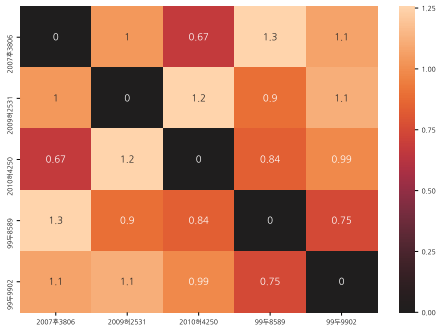


Figure 4-33: Similarity of Input Nodes for CLC

Figure 4-34: Heatmap for W_2 Matrix for CLC

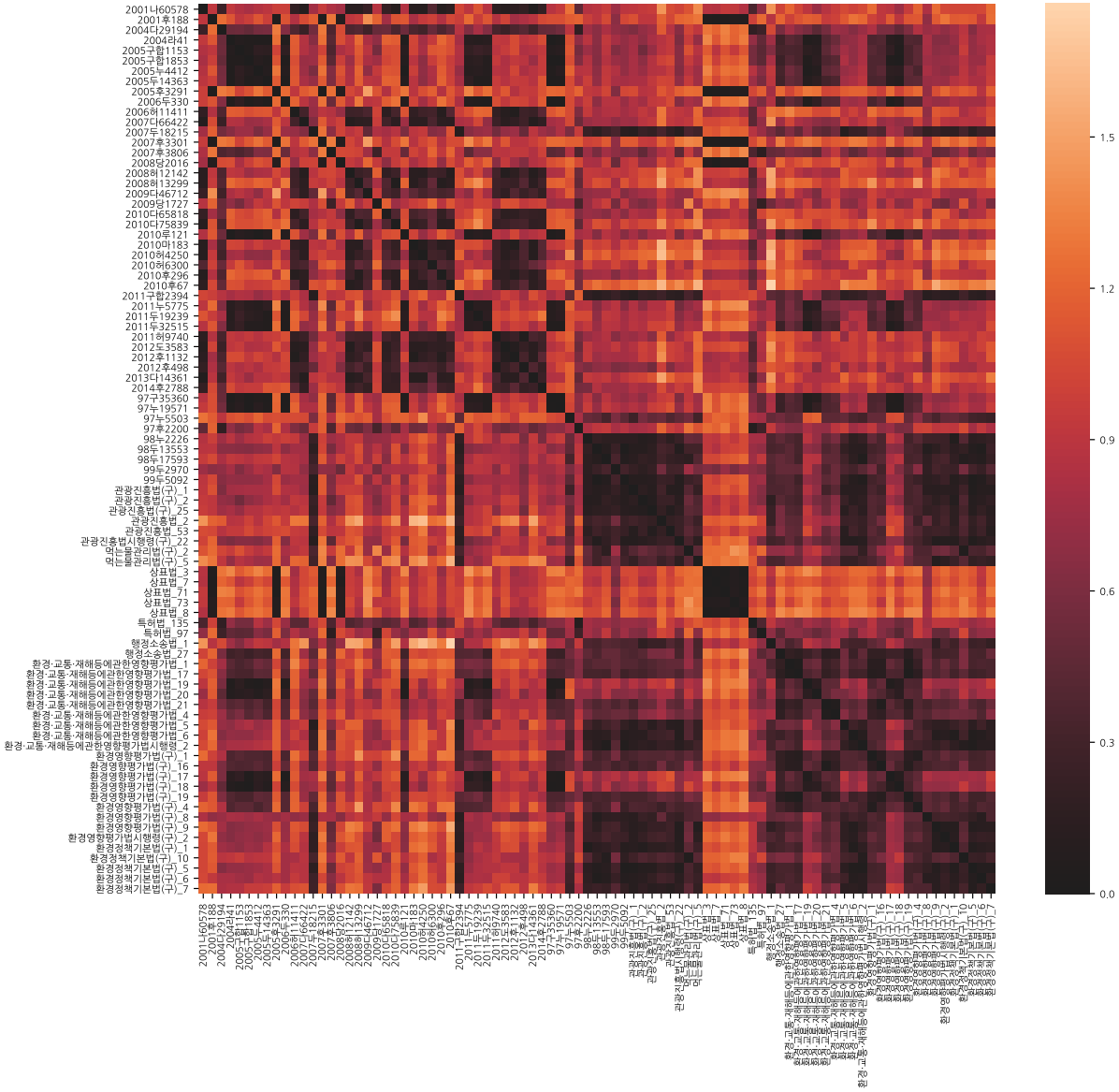


Figure 4-35: Similarity of Output Nodes for CLC

4.3.4.5 Comparison of model architectures

We now collect and compare all the results. The similarity for the answers of each model is drawn out in Table 4-7. We have also included similarity results that were calculated using a word-based method in a TF-IDF structure.

With the word-based method, a reasonable similarity value of 0.42 was calculated for case 2009허2531 and 2010허4250. However, the similarity value of case 99두8589 and 99두9902 came out as 1.3 wrongly indicating that they have no similarity at all. This is because the keyword match between the two cases are relatively low and this is shown in Table 4-7. Through these results we can determine that learning based on legal citation relationships performs much better than previous word-based methods.

	Similarity			
Gold Standard Pair	CL	CC	CLC	Word
2007후3806 , 2010허4250	0.68	0.68	0.67	0.42
99두8589 , 99두9902	0.91	0.89	0.75	0.81
2009허2531	-	-	-	-

Table 4-7: Similarity for Law2Vec models

(0 near best)

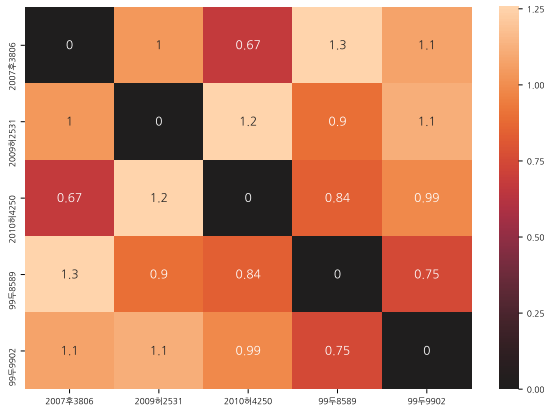


Figure 4-36: Similarity of Input Nodes for CLC

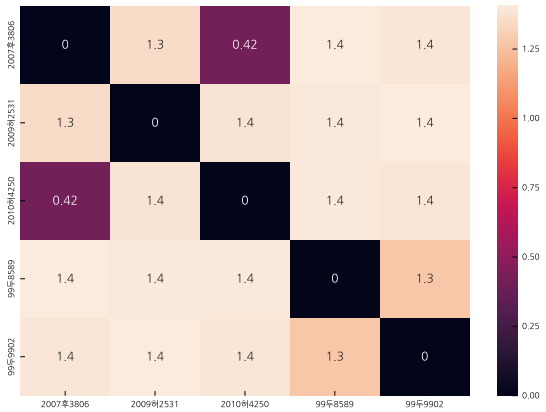


Figure 4-37: Similarity of Input cases for Word

4.4 Link Prediction

To apply our proposed model to actual services and products, we underwent learning with a large-scale set. Learning data was composed of 148,325 cases and 47,444 legislation sections and

was structured according to Table 4-8. To evaluate the performance, we had an answer set of 300 cases that were especially picked out (by a lawyer) for their similarity.

	# of Relations	# of Learning Set
CL	73,211	326,136
CC	145,100	427,793
CLC	148,325	753,929

Table 4-8: Legal Data Description

4.4.1 CL Model for large-scale set

The learning process can be seen in Fig 4-38 and 4-39. When the CL Model was used for learning, we could see that the answer sets gave out very good results. The answer distribution at iteration 100 was near 0 whilst at iteration 35,800 the answer distribution was closer to 1.

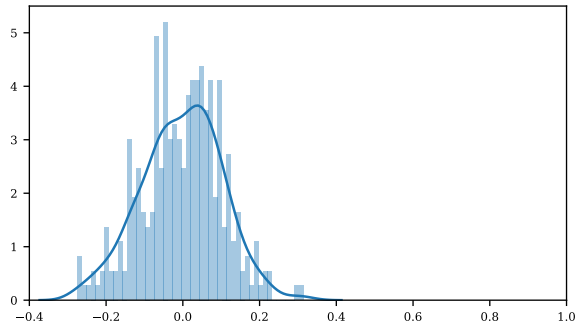


Figure 4-38: Answer Distribution for CL (1 is best) at iteration 100

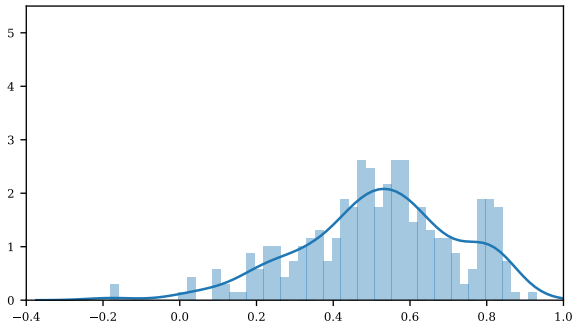


Figure 4-39: Answer Distribution for CL (1 is best) at iteration 35800

4.4.2 CC Model for large-scale set

With the CC Model, learning did not go well (Fig 4-40, 4-41). We interpret this to be because the answer set contained data that were not largely affected by case citation relationships.

4.4.3 CLC Model for large-scale set

Due to the large amount of learning data, we could see that the learning progress was much slower for the CLC Model. Though slow, as the learning processed, we could see that the answer distribution got closer to 1 (1 is best, Fig 4-42, 4-43).

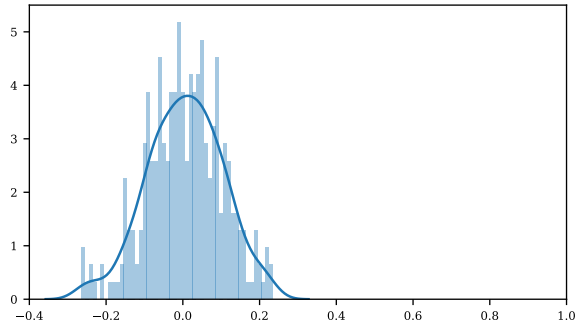


Figure 4-40: Answer Distribution for CC (1 is best) at iteration 100

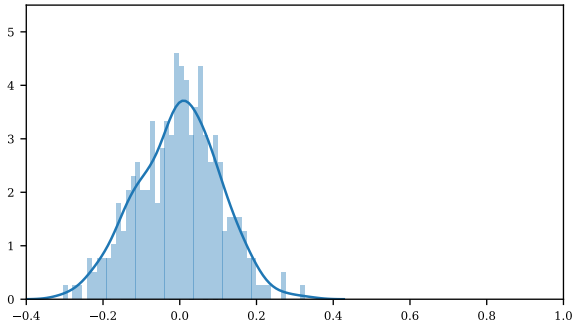


Figure 4-41: Answer Distribution for CC (1 is best) at iteration 27100

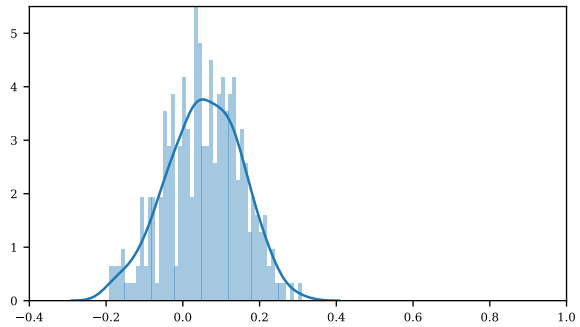


Figure 4-42: Answer Distribution for CLC (1 is best) at iteration 100

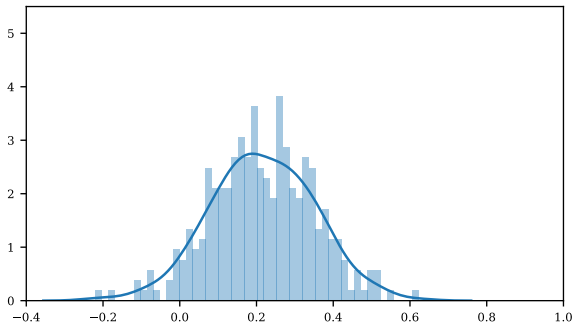


Figure 4-43: Answer Distribution for CLC (1 is best) at iteration 1690

4.5 Results and discussion

Law2Vec calculates the semantic similarity of documents excellently compared to previous word-based methods. This is because word-based methods cannot solve the dismissal of synonyms, variants and other characteristics of language whilst Law2Vec wonderfully takes in cited legislation sections and cases to identify the content of a case that involves material facts of the case and a judge's reasoning.

The advantage of learning based on citation relationship is not only that the similarity of content of each document can be calculated, but also that we can predict the content of publicly undisclosed cases. In Korea, hundreds of judgments are made yet only a minority of these are released and disclosed to the public due to the Privacy Act that wills for the personal details of appellants and respondents to be protected. Currently, only 77,000 cases have been made public and 148,000 cases if we consider the cited cases of these public cases. What this means is that we are not able to access the content of the undisclosed 70,000 cases. With Law2Vec, because we can predict the similarity of cases using the calculation of semantic similarity through the learning of citation relationships, we are able to predict the content of the undisclosed cases.

Moreover, as we saw in our experiment, using matrix W2, we can calculate the semantic similarity of legislation sections. The context of legislation sections have been written to well

contain a specific meaning allowing each section to have concise context. This means that with a word-based method, it would be hard to analyze the meaning and written context of each section of legislations. However, through the learning of citation relationships of legal cases, we are able calculate legislation similarity and obtain the advantage of calculating semantic similarity between sections.

To summarize, Law2Vec has the following advantages :

- Is a learning method specialized for the structure of legal data (that is composed of legal reasoning and material facts)
- Shows superior performance for semantic similarity calculation compared to traditional word-based method
- Can calculate semantic similarity of undisclosed cases
- Can calculate semantic similarity of legislation sections

5

Applications

During the course of researching Korean text mining, we were able to produce some interesting applications. We would like to introduce two of them.

5.1 News Summarization System

Our News Summarization System extract the three summarized texts (Fig 5-1). Input your selected news articles to the blank text fields in the left side, then click the “Summ” button. And it will split the sentences and calculate the importance score of each sentence. Try it on the site `http://summ-dev.ap-northeast-2.elasticbeanstalk.com/`.

Figure 5-1: Demo of News Summarization System

5.2 Link prediction for Legal Data

Based on the result of Law2Vec, we can provide a link (relation) prediction service as in Fig 5-2. A link prediction will link a certain data (case) to other relevant data with a green line. As shown in Fig 5-2, the selected case in coloured in green will be linked to cases that have no direct citation relationships with the selected case but are similar in content. Therefore, semantically similar cases can be predicted with such method. You can try it on the site `http://lawbot.org`.

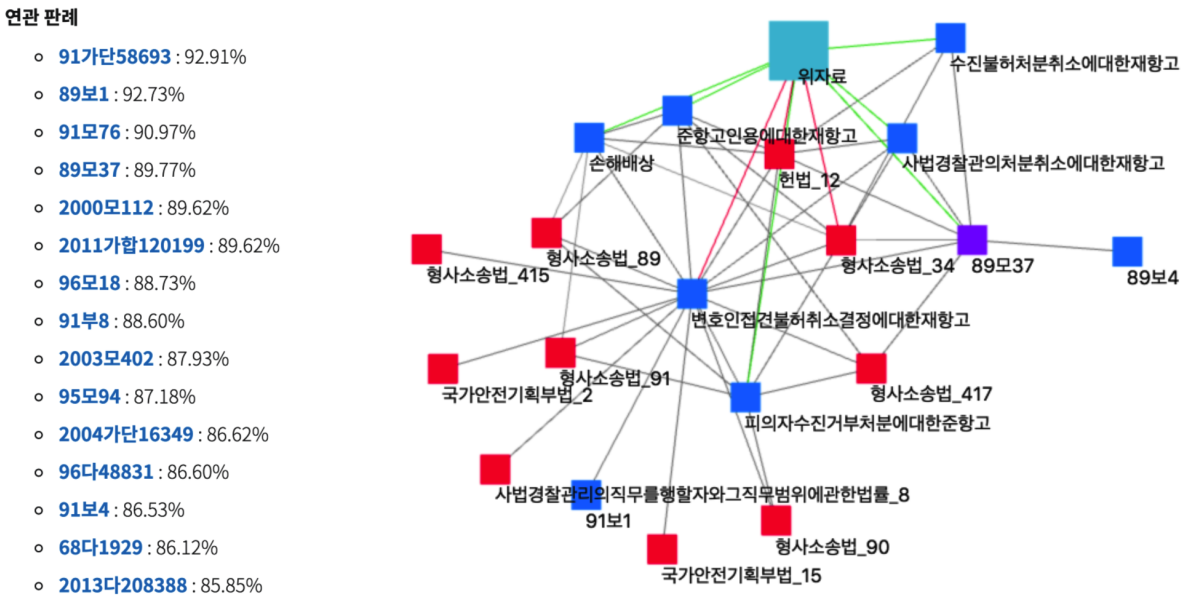


Figure 5-2: Demo for Legal Data Link Prediction using Law2Vec

References

[1] Huberman, Bernardo A., and Lada A. Adamic. "Internet: growth dynamics of the world-wide web." Nature 401.6749 (1999): 131-131.

[2] Weiss, Sholom M., et al. Text mining: predictive methods for analyzing unstructured information. Springer Science & Business Media, 2010.

[3] Zadeh, Lotfi Asker. "Fuzzy sets." Information and control 8.3 (1965): 338-353.

[4] Yager, Ronald R. "A new approach to the summarization of data." Information Sciences 28.1 (1982): 69-86.

[5] Niewiadomski, Adam. "News generating via fuzzy summarization of databases." SOFSEM 2006: Theory and Practice of Computer Science. Springer Berlin Heidelberg, 2006. 419-429.

[6] Kiani, B. A., T. M. R. Akbarzadeh, and M. H. Moeinzadeh. "Intelligent Extractive Text Summarization Using Fuzzy Inference Systems." Engineering of Intelligent Systems, 2006 IEEE International Conference on. IEEE, 2006.

[7] Suanmali, Ladda, Naomie Salim, and Mohammed Salem Binwahlan. "Fuzzy logic based method for improving text summarization." arXiv preprint arXiv:0906.4690 (2009).

[8] Gupta, Vishal, and Gurpreet Singh Lehal. "A survey of text summarization extractive techniques." Journal of Emerging Technologies in Web Intelligence 2.3 (2010): 258-268.

[9] TF-IDF — Wikipedia, The Free Encyclopedia, `http://en.wikipedia.org/wiki/Tf%E2%80%93idf`

[10] Kyoomarsi, F., Khosravi, H., Eslami, E., & Davoudi, M. "Extraction-based text summarization using fuzzy analysis." Iranian Journal of Fuzzy Systems, 7(3), 15-32. (2010)

[11] Benes, Peter. "Bachelor's Thesis.", `http://users.fs.cvut.cz/ivo.bukovsky/PVVR/prace_studentu/Fuzzy_Crane_Benesh_unnoficial.pdf`

[12] `http://peach.googlecode.com/hg/doc/build/html/index.html`

[13] Suanmali, Ladda, Naomie Salim, and Mohammed Salem Binwahlan. "Fuzzy genetic semantic based text summarization." Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on. IEEE, (2011)

[14] Wang, Jiabing, Hong Peng, and Jing-song Hu. "Automatic keyphrases extraction from document using neural network." Advances in Machine Learning and Cybernetics. Springer Berlin Heidelberg, 2006. 633-641.

[15] Matsuo, Yutaka, Yukio Ohsawa, and Mitsuru Ishizuka. "Keyworld: Extracting keywords from document s small world." Discovery Science. Springer Berlin Heidelberg, 2001.

[16] Berry, Michael W. "Survey of Text Mining" Springer. 2004.

[17] Hangul — Wikipedia, The Free Encyclopedia, `https://en.wikipedia.org/wiki/Hangul`

[18] Hu, Tao, and Tu Peng. "Multi-angle Evaluations of Test Cases Based on Dynamic Analysis." International Conference on Advanced Data Mining and Applications. Springer, Cham, 2014.

[19] Kim, Jin-Suk and Choe, Ho-Seop and You, Beom-Jong and Seo, Jeong-Hyun and Lee, Suk-Hoon and Ra, Dong-Yul. "HKIB-20000 & HKIB-40075: Hangul Benchmark Collections for Text Categorization Research" Journal of Computing Science and Engineering 3.3 (2009): 165-180.

[20] Salton, Gerard, Anita Wong, and Chung-Shu Yang. "A vector space model for automatic indexing." Communications of the ACM 18.11 (1975): 613-620.

[21] Simovici, Dan A. Linear algebra tools for data mining. World Scientific, 2012.

[22] Aggarwal, Charu C., and ChengXiang Zhai. "A survey of text clustering algorithms." Mining text data. Springer US, 2012. 77-128.

[23] Huang, Anna. "Similarity measures for text document clustering." Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. 2008.

[24] Mohler, Michael, and Rada Mihalcea. "Text-to-text semantic similarity for automatic short answer grading." Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009.

[25] Trefethen, Lloyd N., and David Bau III. Numerical linear algebra. Vol. 50. Siam, 1997.

[26] Dumais, Susan T., et al. "Using latent semantic analysis to improve access to textual information." Proceedings of the SIGCHI conference on Human factors in computing systems. Acm, 1988.

[27] Deerwester, Scott, et al. "Indexing by latent semantic analysis." Journal of the American society for information science 41.6 (1990): 391.

[28] Manning, D. A. C. "Introduction to information retrieval." Introduction to Industrial Minerals. Springer Netherlands, 1995. 1-16.

[29] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." Journal of machine Learning research 3.Jan (2003): 993-1022.

[30] Regan, D., and S. K. Srivatsa. "A NOVEL SENSING NOISE AND GAUSSIAN NOISE REMOVAL METHODS VIA SPARSE REPRESENTATION USING SVD AND COMPRESSIVE SENSING METHODS." (2006).

[31] Halko, Nathan, Per-Gunnar Martinsson, and Joel A. Tropp. "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions." SIAM review 53.2 (2011): 217-288.

[32] Kambatla, Karthik, et al. "Trends in big data analytics." Journal of Parallel and Distributed Computing 74.7 (2014): 2561-2573.

[33] Lee, Daniel D., and H. Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization." Nature 401.6755 (1999): 788-791.

[34] Lee, Daniel D., and H. Sebastian Seung. "Algorithms for non-negative matrix factorization." Advances in neural information processing systems. 2001.

[35] Xu, Wei, Xin Liu, and Yihong Gong. "Document clustering based on non-negative matrix factorization." Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003.

[36] Kim, Hyunsoo, and Haesun Park. "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis." Bioinformatics 23.12 (2007): 1495-1502.

[37] Amine, Abdelmalek, Zakaria Elberrichi, and Michel Simonet. "Evaluation of text clustering methods using wordnet." Int. Arab J. Inf. Technol. 7.4 (2010): 349-357.

[38] Zhai, ChengXiang, and Sean Massung. Text data management and analysis: a practical introduction to information retrieval and text mining. Morgan & Claypool, 2016.

[39] Bezdek, James C., and Nikhil R. Pal. "Cluster validation with generalized Dunn's indices." Artificial Neural Networks and Expert Systems, 1995. Proceedings., Second New Zealand International Two-Stream Conference on. IEEE, 1995.

[40] Rousseeuw, Peter J. "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis." Journal of computational and applied mathematics 20 (1987): 53-65.

[41] Caliński, Tadeusz, and Jerzy Harabasz. "A dendrite method for cluster analysis." Communications in Statistics-theory and Methods 3.1 (1974): 1-27.

[42] Van Rijsbergen, C. J. "Information Retrieval." (1979).

[43] Pfitzner, Darius, Richard Leibbrandt, and David Powers. "Characterization and evaluation of similarity measures for pairs of clusterings." Knowledge and Information Systems 19.3 (2009): 361-394.

[44] Sokolova, Marina, Nathalie Japkowicz, and Stan Szpakowicz. "Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation." Australian conference on artificial intelligence. Vol. 4304. 2006.

[45] Sasaki, Yutaka. "The truth of the F-measure." Teach Tutor mater 1.5 (2007).

[46] Metadata — Wikipedia, The Free Encyclopedia, `https://en.wikipedia.org/wiki/Metadata`

[47] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).

ACKNOWLEDGEMENTS

Is there such a thing as coincidence in this world? Everything that happens is the result of randomness resulting from our choices. What about my job, my colleagues, or the people I love? I don't believe these are coincidences. And quite naturally, this work is also not merely a coincidence. Many people have helped me and given me what I needed to make it a reality.

It's not by chance that everything worked out so well. There were many options for me in this world, including some that would have been dangerous, hard, or painful. Thus, I would like to thank those many of you who led me toward the right choices among all the possibilities. Every day, I make prayers of thankfulness for the people I love.

I'm grateful for my parents and my brother, my greatest supporters of all.

I'm grateful for my professors. They are leaders who lead me to seek wisdom and gain enlightenment.

And I'm grateful for my friends, my lifelong companions and powerful forces in my life.

To arrive where I am now, I have been helped by so many people. I cannot imagine myself without any of them. I'm also grateful for all of these people.

No one exists by himself, and I am here because of others. I will spend my life trying to return what I have been given and share it with others as well.